JPRS-UCC-90-006
12 SEPTEMBER 1990

FOREIGN BROADCAST INFORMATION SERVICE

JPRS Report

Science & Technology

USSR: Computers

19981217 122

JPRS-UCC-90-006
12 SEPTEMBER 1990

SCIENCE & TECHNOLOGY

USSR: COMPUTERS

CONTENTS

UDC 681.3.06:519.68

Basic-Structural User-Shared Systems Design Technology

907G0042A Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 1989 pp 42-44

[Article by A. I. Khalilov]

[Text] Given the increasingly rapid development of scientific and technical progress, the reconstruction of economic management based on economic methods and the extensive computerization of all spheres of human activity the issues of improving the utility of computer technology from microprocessors through supercomputers have become especially significant.

The mass manufacturing of personal computers (PCs) has established the conditions for effective solution of many classes of operational problems using local data bases (DB) which could not be implemented on mainframe computers due to their high cost, unreliability, remote location, etc.

At the same time personal computers have not displaced commercial data processing systems (IVS) based on mainframe computer technology and representing the basic element of the information processing industry. On the contrary, their role in computerization of both the elements generated by the information technology of solving important problems based on large data bases and simulating large agricultural or scientific and technical facilities and processes (regional management systems, economic and meteorological systems, transport, energy, and industrial facilities and systems etc.) and of universal user-shared systems (SKP) have become more clearly defined.

The extensive architectural capabilities, the wide variety of users and problems, on the one hand, and the use of various tools and software, on the other, have produced three interactive components in actual user-shared systems: $\{\sigma\}$ - the set of user shared system software tools; {M} - the subject domain set; $\{\Omega\}$ - the operating environment set in which $\{\sigma\}$ operates on {M}.

In order to improve the efficiency and ease of manufacture of user-shared systems and support the implementation of the capabilities discussed above it is necessary to minimize the set $\{\sigma\}$ (integration) with a maximum size of {M} (adaptability of to M) and guarantee invariance of σ with respect to

Ω. In this formulation the independently important problem of improving the speed of each individual computer installation is not the determinant problem. In addition the problem of optimum organization of the computational process to support both a flexible and a dynamic handling discipline is also a timely problem.

Here the discussion concerns a system whose structure is in the general case a multilevel multiprocessor inhomogeneous structure.

The composition and characteristics of problems and users are not predefined. The problems may extend over a broad spectrum: from complex iterative calculations which will involve many system components at different hierarchical levels, through simple data base requests. Clearly such a user-shared system must have a number of specific properties: flexibility and extensive adaptability, reliability, and rapid response.

The user-shared system may include different computers integrated into multiprocessor systems, while the network architecture must be capable of supporting distributed information processing as well as interaction between not only the user and the system but also between the system elements, i.e., it must have the property of communicability.

Whereas for an individual computer installation the problem is one of improving computer speed by modernizing algorithms, programs, data structures, or other elements, this is insufficient for a user-shared system. Organization of the computational process, definition and dynamic support of optimum operation of the overall computer complex accounting for the multiple internal and external parameters are the most important elements here. A user-shared system must be able to respond quickly to changes in its parameters, i.e., it must be a parametric system.

The wide variety of users assumes that the user group will include individuals with the broadest range of levels of computer literacy: from system programmers through the "computer illiterate." The system must be capable of servicing all subscribers on a high level, provide the capacity for contributing user intelligence to the resulting product and best use of user knowledge, capabilities, instinct, habits, experience, etc. The user-shared system must have a rather high intelligence level.

Therefore the solution to the problem of user sharing of software and hardware resources requires the development of a parallel interactive user-shared system (ISKPPD) which combines within a single technology the scientific and technical results from solutions to the problems outlined above and having the properties of communicability, intelligence, and parametric operating capability.

The methodology behind the development of user-shared systems assumes integration within a single technological philosophy of such concepts as the principles of system structural design [1] and the successive analysis method [2], interactive system software and design tools [3], data [4] and knowledge representation methods, interactive multiprocessing and simulation [5], etc.

Underlying this technology are the principles of overall system and constituent component structuring and the concept of data base design; this technology has come to be called the basic-structural technology (SBT).

This technology has reflected the development of the idea of structural design and programming as well as the decomposition of organizational structures aimed at maximum identification of control processes, control process integration and universal implementation of algorithms, identification in the subject domain model of the object data base, environment data base, relational data base, and rules bases for altering these relations. This approach is fundamentally new and provides the basis (in combination with interactive design techniques and tools for developing knowledge bases (BZ) and their associated intelligence systems for broad application.

The primary analysis, initial structuring and dynamic restructuring methods for the entire system as well as for its structural and engineering elements (control system, subject domain model, data bases, multiprocessing system, interactive schemes, etc.) employ the successive analysis method (MPU) which is based on existing tree processing methods and algorithms and is differentiated from such algorithms and techniques by its generality, universality and functional simplicity.

The data base design concepts are applied by means of frame technology to the functional system software which is responsible for the capability to generate program data bases and associated management systems similar to data bases and data base management systems. The concept of a data multibase (MBD) for application to parallel user-shared data bases is introduced together with the definition of a relational data multibase and the generalized data base manipulation operations.

The tools to develop interactive systems (DS) by extensive use of frame technology are differentiated by universal control and management capability and simple adaptation to different subject domains as well as the capability to develop distributed hierarchical data bases and knowledge bases by combining the structuring, successive analysis method, and interactive tools outlined above.

A functional diagram of the interactive multiprocessing system is formulated; this system is based on new serial program parallelling algorithms by means of serial analysis in addition to the techniques and tools outlined above, which has made it possible to obtain a complete parallel form of the program on this language level together with program optimization methods and also permits the use of other existing and future program parallelling techniques and methods.

The use of the successive analysis method as well as optimization techniques, weight functions, and other methods in the basic structural technology makes it possible to obtain parallel programs that are optimized for different criteria: from maximum parallel programs through strictly serial programs.

The approach to modeling the user-shared system used in the basic structural technology contains fundamentally new elements. Such elements involve expanding the use of existing methods of simulating the engineering characteristics and functional modes of the system to include modeling of such characteristics in combination with economic planning and other properties and optimizing these factors based on the use of data bases and knowledge bases containing information on the functional processes, users, environment and operational conditions of the user-shared system as well as the normative technological operation base.

The primary design concepts underlying the basic structural technology are implemented within the framework of the DISUM user-shared interactive system designed for computer-aided design of applied interactive systems and for management of user-program module interaction as well as user-data base interaction in the user-shared mode.

The DISUM user-shared system is used to develop applied interactive systems as a standard control component and hence its range of application is determined by the function of the applied software of the simulated subject domain.

DISUM is oriented for users in the following categories: applied programmer, interactive system designer, end user.

The applied programmer can use interactive tools as well as accessing tools to access a given subject domain.

An interactive system designer can set up interaction for a user using a unified technology, can use the tools to describe interaction and objects involved in the interaction (program modules, menus, formats, "prompts", system messages, etc.) as well as software tools for developing and debugging the interactive systems under design.

The DISUM provides the following for the end user:

- access to the information and program resources of the interactive systems accounting for user priority and ability to use interactive system capabilities;

- interactive problem solving using a unified technology independent of the problem orientation of the tasks;

- a set of interactive mode service functions (list generation, establishment of protocols for interaction or interaction sessions, utilization of a cataloged sequence of responses, etc.);

- techniques for accessing subject domain data (logical condition search and retrieval, correction, addition or deletion of data), which can be used without special preparation or training.

The DISUM model, like the user-shared system, was designed using the basic structural technology which includes the system process, functional process, and functional program module levels.

The system process level consists of three components having the status of individual YeS OS operating system tasks: the functional process management component (KUFP), the data management component (KUDN) and the terminal management component (KUT).

The primary function of the functional process management component is to distribute the computer and information resources of the interactive systems among the parallel interactive processes. This component is implemented using the PRIMUS applied program package.

The data management component runs data manipulation commands for data arriving in accordance with the functional programs of the interactive system. The SPEKTR data base management system in the group mode (the SPG mode) is used as the data management component.

Standard message management programs (PUS) from the PRIMUS 2.3 applied program package or the PUS based on the common telecommunications access method (TCAM) of the OS YeS operating system and associated with the PRIMUS applied program package nucleus are used to perform terminal management component functions. Message management programs based on TCAM can also be used as interaction tools for interacting with the OS YeS tasks run in other zones.

User-computer interaction which is classified as a subtask of the functional process management component is considered to be an object of the functional process level. An individual copy of the interactive process corresponds to each active interactive system user. The control program of the process is an interactive monitor (DM): a set of universal programs that support interaction using specific configurations. The interactive configuration is a multilayered homogeneous network of framelike elements describing individual steps (states) of interaction as a certain sequence of elementary actions involved in providing such interaction.

The functional module is the object of the DISUM functional program module level; such modules function as media to carry knowledge on the subject domain of the interactive system and implement the functions of the applied interactive system itself. One configuration using an invariant tabular interface (ITI) designed for supporting terminal access to the data base under the control of the SPEKTR data base management system for users with no special training in computer technology is one example of an interaction configuration developed in accordance with the interactive organization concepts using DISUM.

DISUM implements a special interactive program set that makes it possible to generate, modify, and debug the interactive configurations of the resulting interactive
systems.

DISUM can run on any YeS computer model that supports the OS YeS version 6.1 or above operating system in the MVT or CPM modes. The system is oriented for simultaneous support of up to 16 users employing the YeS-7920 terminals.

In this case 350 Kbytes of RAM is required for DISUM when standard software is used. The memory capacity required for normal operation of the applied interactive system is determined by the characteristics of the applied software.

The original DISUM section is based on the OS YeS Assembler language (approximately 7000 instructions), while the message management programs are written in PL/1 (approximately 25,000 statements).

The system response time is 1-3 seconds, while the information recovery time from the data base on the logic level is up to 5 seconds.

The DISUM user-shared system is employed in developing regional control and management systems, supporting user-shared data processing systems, for configuring training and teaching systems, etc.

REFERENCES

1. Khalilov, A. I. "Sistemnyye vzaimodeystviya v ASU territorialnymi zvenyami" [System interactions in territorial node automated management systems], Kiev, 1980.

2. Khalilov, A. I. "The successive analysis method and certain of its applications" Teoriya i praktika sistemnogo programmirovaniya: Sb. nauch. tr. IK AN USSR, Kiev, 1976, pp. 180-191.

3. Khalilov, A. I., Shilkin, A. I. "One approach to the design of user-shared interactive systems", Avtomatizirovannyye sistemy nauchnykh issledovaniy kollektivnogo polzovaniya: Sb. nauch. tr., Moscow, 1984, pp. 38-44.

4. Khalilov, A. I., "Certain issues of data base organization in complex automated control systems", Kibernetika, 1981, No. 3.

5. Khalilov, A. I., Losevichev, V. I., Shutkov, I. M. "Modeling the control component in one user-shared system", Modeli i sistemy obrabotki informatsii: Sb. nauch. tr., Kiev, 1988, No. 7, pp. 97-104.

UDC 65.012.122

Conceptual Model of an Automated Workstation

907G0042B Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 1989 pp 46-48

[Article by L. K. Golyshev]

[Text] The increasing rate of computerization of various spheres of human activity has made it necessary to obtain a specific definition of the concept "automated workstation" (ARM). In domestic publications an ARM is treated as a specifically adapted management workstation (RSU) [1, 2] based on a certain set of specific engineering support subsystems.

The definition of automated workstations may incorporate the general concepts deriving from its conceptual model as examined below; this model represents the result of the natural development and generalization of the workstation (RM) concept: a generally-recognized critically important element in civilized society.

The initial concept of this unit - the workstation - is defined as a localized functionally complete set of hardware and software supporting efficient operator function within a specified time period (working time). The employer will provide this collection of means which includes the tools of labor and the individual means for establishing a comfortable productive working environment, to a specific worker in order to generate high yield labor, support a positive psychological atmosphere during working time and satisfy labor safety conditions.

The system formed by the workstation and the worker can be considered to be an algorithmic system [3] which makes it possible to interpret a specific class of "production" algorithms. This class of algorithms is a special subset of a certain universal set of all special algorithms given the specialization of workstations which has occurred during the integration of technology and labor.

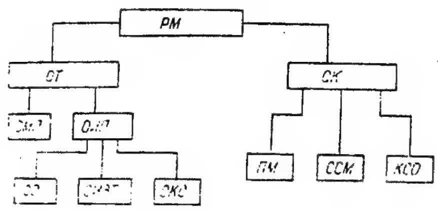


Fig. 1. Generalized workstation structure: OT - tools of labor; CK - environment for maintaining comfortable production conditions; OМП - material production tools; OИП - information production tools; CO - office equipment; CИВТ - information and computer technology; CKC - special communications systems; ПМ - industrial furniture; CCM - microenvironmental systems; KCO - general communications support

Two fundamental algorithm subclasses can be identified within this algorithmic system: the production operation (RPO) subclass and the management operation (RAU) subclass. Each algorithm subclass has specific initial information object description languages (YaSOII), information reprocessing statement sets (NOPI), and syntactic facilities (SS) together with the correspond interpretive facilities (IS).

The set of possible workstation characteristics as a local area (yet not autonomous) system includes factors M1 and M2 - the mechanization and automation levels of labor object and management processing operations, respectively:

$$M1 = \text{ПМА}/\text{П};$$
$$M2 = \text{KMA}/\text{K};$$

where П, ПМА is the measure of the total and the automated volume of industrial operations, respectively; KMA, K is the measure of the necessary management operations and their automated fraction, respectively.

The energy costs of performing manual operations or the resources required for complete automation can be used one measure to define M1. This factor characterizes the "capital intensity" of manufacture.

The labor outlay for manual performance of the operations (for K) as well as the volume of such operations performed automatically (for KMA) can be used as the measure for defining M2. This factor characterizes the "capital intensity" of information production.

It should be noted that the definitions of П and K contain significant ambiguity associated with the difficulty of accounting for the constant change of both types of algorithms resulting from scientific and technical

progress in manufacturing and management. In the general case $П = П(T)$, $K = K(T)$, while it is best to assume that the functions M1(T) and M2(T) are monotonically increasing functions ($M1(T) \longrightarrow 1$, $M2(T) \longrightarrow 1$) whose limits can be achieved by automating the corresponding workstation functions during the corresponding period and by eliminating individual unautomated functions and transferring them to newly created workstations.

A workstation for which the condition M1 = M2 = 1 is satisfied is called a robot.

The workstation information model (Figure 2) can be correlated with a configuration reflecting the composition of the individual workstation components (Figure 1).

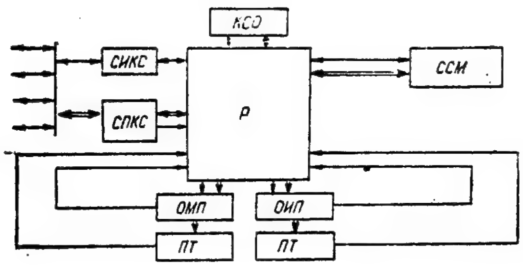


Fig. 2. Structure of workstation information model:
Р - worker; ПТ - objects of labor; СИКС - special information communications systems; СПКС - special productive communications systems; ═══ - material (productive) connections (flows); ──── information interconnections (flows)

An analysis of the model structure makes it possible to identify the control loops in which certain control processes and resulting information flows are implemented.

Each workstation can be set to correlate with a specific rank in the global management process within the workstation system supported by the given automated workstation: strategic management (rank 1), tactical management (rank 2), and operational management (rank 3), together with the industrial support of information flows: information activities (rank 4).

The first three ranks of processes are involved with decision making and their algorithms have unique features.

Standard workstations can be classified by their role in the decision making process as manager workstations (RMR), specialist workstations (RMS), and technician workstations (RMTI).

Introducing the concept of workstation rank and type permits classification of the workstations and analysis of the automated functions.

An analysis of this model makes it possible to identify the semantic relations as well as the volume and time characteristics of the corresponding information flows and to develop the requirements on automation systems.

Therefore satisfaction of a certain functionally-complete set of structured information conversion procedures and implementing each such procedure by means of a hardware-software module - a terminal - can be used as the basis of the concept of workstation automation.

If we consider the automated workstation to be the object of automation the isolated modules (Figure 1) will include units associated with the material production and productive communications functions. These modules are implemented as special automated industrial terminals.

The remaining units comprise an object whose automation capabilities will also be defined as an automated workstation (Figure 3).

Obviously the widely held view that an automated workstation is a program product created solely for computers, primarily personal computers, will undergo changes given that standard computer equipment no longer satisfies the extensive range of requirements on such terminals.

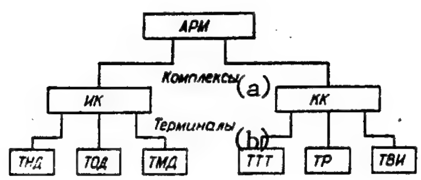


Fig. 3. Automated workstation structure: a - complexes, b - terminals; ИК - information complex; КК - communications complex; ТНД, ТОД, ТМД - data storage, processing, and manipulation terminals, respectively; ТТТ - teletex terminal; ТР - speech terminal; ТВИ video terminal

REFERENCES

1. Kochetkov, G. B. "Avtomatizatsiya kontorskogo truda v SShA" [Design automation in the US], Moscow, 1985.

2. Ioffe, A. F., "Personalnyye EVM i organizatsii upravleniya" [Personal computers in control], Moscow, 1988.

3. Glushkov, V. M., "Vvedeniye v ASU" [An introduction to automated management systems], Kiev, 1972.

UDC 681.51

Intelligent Automated Simulation Systems

907G0042C Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 1989 pp 49-53

[Article by A. A. Lavrov, L. S. Yampolskiy, and V. I. Kostyuk]

[Text] Simulation automation makes it possible to improve analysis, design, and management effectiveness of complex industrial systems. Insufficient formalizability, large size, and complexity of associated problems and the need for extensive user training and the blurring of raw data and criteria in many cases has made it necessary to integrate simulation systems with technical intelligence simulation systems (STII), expert systems (ES), logic calculation system, etc. A significant number of studies in this field have been devoted to the development of intelligent media for developing models, on the use of expert systems for creating a natural language interface with the user during the model run stage, for analyzing simulation results and for developing corresponding recommendations. As a rule automation of the experimental management process is limited by the organization of the model runs in the operational cycle or in accordance with a preestablished plan as well as (more rarely) by the need to support runs in accordance with a certain strict optimization algorithm. At the same time a large number of practical problems require expanding the capabilities of the experimental organization mechanism since simulation is, in the final analysis, aimed at determining the combination of parameter values of the simulation object that optimizes a given quality factor. Consequently management of a series of experiments involves organizing such experiments in a manner corresponding to a solution of a complex optimization problem where the simulation model functions as the means for calculating the quality factor. Moreover, both design and, particularly, management problems are solved under conditions of limits imposed on both the number of calculations and the available time.

It is therefore necessary to search for effective methods of automating simulation experiments. An integrated approach involving a combination of STII functions and the analytic retrieval method module (MAPM) is proposed for solving this problem.

The structural component of the automated simulation system (SAIM) implementing the management function for this series of experiments is called the search model (PM). This simulation model (IM) and the search model

comprise the automated simulation model (AIM). The search model is represented by the analytic search method and/or procedures implemented by the STII. The STII simulation experiment management functions can be divided into two groups:

1. The groups executed by the search method;

2. Supervisory functions.

The primary purpose of running functions in the first group is effective implementation of the present directed series of experiments. Two possible uses of the STII are possible in this case. In the first case the STII will independently implement the primary search stages: representation of raw data, initiation of model run, processing of results and decision making regarding the progression of the experiments. The prerequisites of such a variant may include: lack of strict analytic solution methods and a strict, formalized formulation together with large problem size, etc. In this case the search method is represented by a certain heuristic algorithm that is run within the framework of the apparatus of implicit or explicit logic, semantic nets, etc.

In the second version the STII will interact with routines generated by the analytic search method. Specifically the tasks of the STII will include reducing the number of iterations based on the use of available knowledge, decision making in situations that are not envisaged by the analytical method, etc.

Essentially the supervisor functions are metaprocedures with respect to the search method algorithms or are carried out in order to support additional conditions and limits. Thus the STII, depending on current information, may select or initiate one of several search methods, match the search model to other elements of the SAIM, maintain consistency with the next set of raw data, make decisions regarding the suitability of the present approximate solution in the case of limited time, etc. The overall structure of the STII functions and its interaction with the search model is shown in Figure 1.

During the interaction of the STII and the procedures included in the analytical methods the search model can be represented in implicit form when it corresponds to a combined algorithm consisting of the analytic algorithm and the information processing routines implemented by the STII. In the general case the STII will participate in implementing (or will completely implement) several different search models relating to the same SAIM. Figure 1 illustrates this case by a partial cross-section of the STII units and a specific search model. It is established that this search model includes the STII resources (time, computational, etc.) that are used in its implementation. Moreover, the STII will address other simulation problems (in this case, model formulation, interactive support) by interacting with the remaining SAIM components.

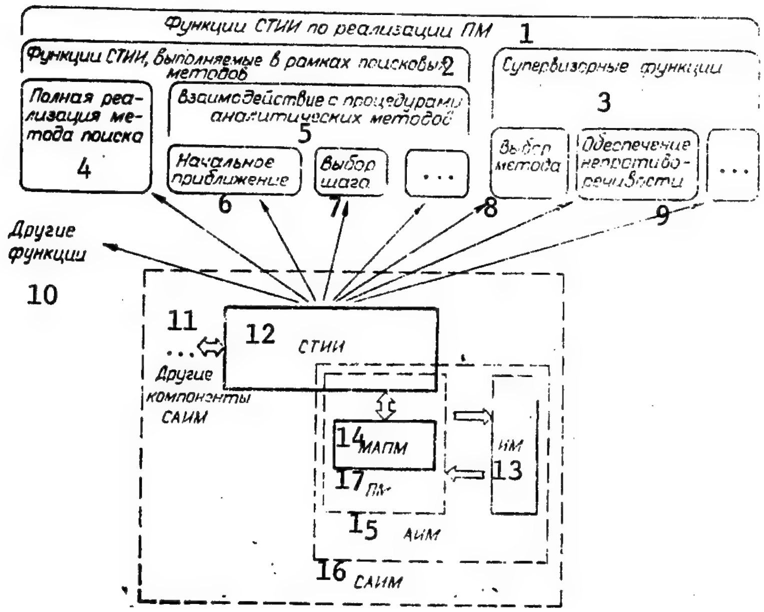


Figure 1. General Structure of STII Functions

Key:

1. Search model implementation functions of the technical intelligence simulation system
2. STII functions carried out within the framework of search methods
3. Supervisory functions
4. Complete implementation of the search method
5. Interaction with analytic routines
6. Initial approximation
7. Step selection
8. Technique selection
9. Consistency
10. Other functions
11. Other automated simulation system components
12. STII
13. Simulation model
14. MAPM
15. Automated simulation model
16. Automated simulation system
17. Search model

STII operation is based on knowledge utilization. The specifics of its practical application (in this case: simulation experiment management) will affect above all the knowledge of the problem domain. Their composition is represented by knowledge of the simulation object, the simulation model, and management of the simulation experiments (the search method, additional conditions and their execution).

Knowledge of the simulation object reflects the essence of the various relations defined in the set of parameters of the simulation object and in the set of values of these parameters. Such knowledge can contain the following information:

- the configuration of optima for several criteria: IF "maximum speed" is required, THEN "set batch size from ... to ...";

- the dynamic interrelationship of parameters: IF "input parameter x_1 increases", THEN "output parameter y_1 diminishes";

- subjective criteria: IF the "production parameters of the machines are identical", THEN "use a more ergonomic machine". (These are arbitrary examples.)

Knowledge of the simulation model contains information on the characteristics of the simulation model and the features of using this model: the oriented run duration, accuracy of results, etc.

Knowledge of the experimental management process can be divided into two classes. The first includes information on the search strategy and/or the heuristics which facilitate improved effectiveness of the analytic techniques. Knowledge in the second class consists of information used to carry out control or other functions; this includes information on time resources, the applicability conditions of a specific method, etc.

The knowledge acquisition stages and sources in the STII are as follows:

- initial accumulation stage; the sources are objective positions deriving from the nature of the simulated object, the search method and the experimental management process; the available and interpreted results from previous experiments; the operational experience of the simulated object and the use of search techniques and simulation methodology;

- the preliminary operating state; the sources include: the interpreted results of experiments carried out using the SAIM, yet obtained prior to the solution of the given problem;

- the present accumulation stage; sources - the interpreted experimental results from the present series; operational information on goals and experimental conditions.

The process of acquiring knowledge on automated experimentation has a number of specific features:

- the simulation model is used as the means for obtaining a substantial portion of the knowledge;

- accumulation and processing of experimental data and knowledge representation based on this data all take place in an automated (automatic) mode;

- during the SAIM development stage a comprehensive detailed calculation experiment is carried out in order to identify the primary principles and of modeling the object to obtain statistically significant results.

The data and knowledge acquired here are stored in the STII. A mechanism is provided for using this a priori information for an effective search during the SAIM operation stage, which makes it possible to improve substantially the utility of the system, particularly under time constraints since the more labor-intensive, routine procedures are partially eliminated from the decision making stage and are implemented in the SAIM development stage. In this case system operation will involve only experiments intended to refine or identify the required information and for final decision making. This approach may be employed when the SAIM is used in real-time control systems.

The interaction between the automated simulation system and the analytic search method module is shown in Figure 2.

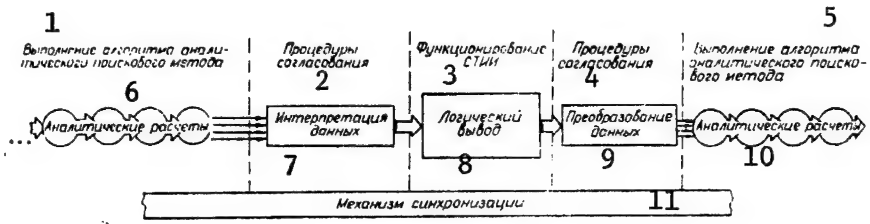


Figure 2. STII-MAPM Interaction Configuration

Key:

1. Implementation of analytic search algorithm
2. Matching procedures
3. STII operation
4. Matching procedures
5. Execution of analytic search algorithm
6. Analytic calculations
7. Data interpretation
8. Logic output
9. Data conversion
10. Analytic calculations
11. Synchronization mechanism

The necessary conditions for implementing interaction include:

- availability of matching procedures which analyze and convert the information status of the automated simulation system and interpret the analytic calculation data in order to generate logic output transmissions and convert numerical data based on logic output results;

- the availability of a synchronization mechanism to provide the necessary alternation, time separation, and/or parallelling of the individual routines (logic output, analytical, matching) carried out in the search model.

An arbitrary example is given below to illustrate the combined approach. Assume the simulation is to be used to determine the value of the parameter x_1 from the range L which maximizes the quality factor f. The STII contains information that the function $f(x_1)$ is unimodal in L. On the basis of this information the STII selects the search method (for example, the Fibonacci method).

In addition, the STII makes an initial approximation or localizes (narrows) the range. The following are knowledge (rules) samples that could be used in this case: IF "the variable parameter is x_1" AND "the remaining parameters are fixed" AND "the range of the parameter x_1 lies from ... to ...", THEN "f is unimodal"; IF the maximum parameter is "f" AND "x_2 lies from ... to ...", THEN "parameter x_1 will be selected in the range from ... to ...".

Another particular example can be used. Assume the range of the parameters is a finite set of points and there is no a priori information available on the form of the response surface; moreover, it is difficult to represent a meaningful practical power relation within this set. It is necessary to optimize a certain factor while providing a comprehensive solution (i.e., it is not possible to reject versions without guaranteeing that they are not valid). The traditional simulation method would involve an exhaustive search of all versions and a complete run of each version. At the same time in the case of the determinate simulation model we can expect a potential reduction in the total analysis time by using estimation functions. For this purpose the model run is interrupted at a certain point and an optimistic forecast of the criterion value is then generated on the basis of intermediate results. If the derived result is worse than that obtained previously, the run is not resumed and the procedure jumps to the next point in the domain.

Obviously the effectiveness of this method is largely determined by the successful selection of the initial point: the closer this point is to the optimum the greater the number of successive versions rejected after each partial run.

The STII will in this case perform the initial approximation.

This example can be used in a specific application. Thus assume it is necessary to minimize the duration of the processing cycle of a set of batches of parts for a given production line. The following could be sample production rules in the STII which will be used as the basis for selecting the initial point: IF "the minimized parameter is cycle duration", THEN "it is necessary to select batches whose labor intensity is minimized for loading the machine"; IF "the minimized parameter is the cycle duration", THEN "it is necessary to uniformly load the machines for the initial start-up"; IF "the minimized parameter is cycle duration" AND "several batches have similar processing routes", THEN "use back-up machines", etc.

Moreover, the STII makes a decision regarding the suitability of using estimation functions and also selects the necessary function from these options. The following types of rules play a significant role here: IF "time is limited", THEN "use estimation functions"; IF "the processing method is a serial method", THEN "select estimation function number —", etc.

If a serial method is used to process the parts media, $T^*_{ц}$ can be used as the estimate of the production cycle duration $T_{ц}$:

$$T^*_{ц} = t_{кт} + \max_i \{((c_i - n_{ik_i}) t_{ik_i} + c_i \times \sum_{j=k_i+1}^{l_i} t_{ij})\},$$

where t_{KT} is the model time at the checkpoint; c_i is the number of media in the i^{th} batch; k_i is the processing stage of the i^{th} batch corresponding to the interrupt point; n_{iki} is the number of i^{th} batch media whose k^{th} processing stage has terminated with the interrupt point; t_{ij} is the processing time of a single i^{th} batch medium in the j^{th} stage in the production route; l_i is the number of stages in the production route of the i^{th} batch. Here $T_{ц} \geq T^*_{ц}$.

Previous "Imitator-1" [Simulator-1] and "Dispetcher-1" [Dispatcher-1] [1, 2] systems employed a traditional exhaustive search of versions and even with limited experimental automation capabilities this made it possible to improve substantially the effectiveness of industrial design and management problem solving. An experimental analysis of the application of the elements of this combined approach to computer simulation confirms the suitability of its use in the "Imitator-2" and the "Dispetcher-2" systems developed by the Technical Computer Science Department of the Kalinin Polytechnic Institute. The heuristic procedures in these systems are based on specially-interpreted strict and fuzzy logic Petri nets which implement propositional logic functions.

REFERENCES

1. Yampolskiy, L. S., Polishchuk, M. N., "Optimizatsiya tekhnologicheskikh protsessov v gibkikh proizvodstvennykh sistemakh" [Optimization of industrial processes in flexible manufacturing systems], Kiev, 1988.

2. Lavrov, A. A., Yampolskiy, L. S., Yampolskiy, S. L., "Real-time planning module with an embedded simulation model", MEKHANIZATSIYA I AVTOMATIZATSIYA UPR, 1989, No. 3.

UDC 681.3.06:519.81

Personal Computer Hardware and Software Selection Based on Use-Requirements

907G0042D Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 1989 pp 53-56

[Article by Yu. S. Yakovlev, Yu. M. Shtern, N. V. Nesterenko, and M. V. Freyman]

[Text] There is a significant increase in the manufacture of various models of domestic personal computers in the USSR today together with an increase in the level of personal computer imports, including imports of peripheral devices, software, etc. [1]. The expanding inventory of personal computer models and associated hardware and software makes it possible to use personal computers as tools for an increasingly broad variety of applications. At the same time as both domestic and foreign experience demonstrates the following factors may serve to reduce the effective introduction and utilization of computer hardware and personal computers to a substantial degree:

- the lack of knowledge of personal computer capabilities for problem solving in each subject domain by many potential users;

- a virtual complete lack of methodological or other materials on personal computer applications (characteristics) as well as associated hardware and software (for example, catalogs);

- a shortage or poor quality and in some cases a complete lack of documentation for popular personal computer software;

- lack of advertisement information etc.

This complicates user efforts to solve problems associated with the acquisition and application of personal computers and personal computer hardware and software as well as professional peripherals [1] for, for example, problem oriented systems or automated workstations (ARM) based on personal computers using professional hardware modules. Hence one of the most critical issues is integration of systems based on personal computers for a specific workstation requirement [2] for which a personal computer-based automated workstation is required; this provides solutions to a number of problems for a wide range of users [3].

The problem involves configuring an automated workstation hardware-software system (sets) or other personal computer-based problem-oriented systems that will satisfy user requirements with respect to applications, equipment functions, software functions, workstation layout, etc. [1, 4].

Let $F = \{F_1, F_2, \ldots, F_f\}$ represent the set of user functional requirements; $T = \{T_1, T_2, \ldots, T_t\}$ represent the set of workstation requirements; $A = \{A_1, A_2, \ldots, A_a\}$ represent the set of personal computer hardware characteristics; $P = \{P_1, P_2, \ldots, P_p\}$ represent the set of software characteristics; $C = \{C_1, C_2, \ldots, C_c\}$ represent the set of personal computer structural characteristics; $M = \{M_1, M_2, \ldots, M_m\}$ represent the set of personal computer models; $B = \{B_1, B_2, \ldots, B_b\}$ represent the set of personal computer peripherals and units; and $S = \{S_1, S_2, \ldots, S_s\}$ represent the set of personal computer software components.

The elements of these sets are called objects, while the sets themselves are called object groups [4].

The user selects the object set from different groups; as a rule, from groups F and T. It is necessary to define object sets from groups M, B, and S that are satisfactory for a number of criteria accounting for the relations between the objects in the different groups. The object inventory and object links as well as other necessary characteristics can be determined initially (for a certain time) by expert analysis.

The solution to the problem involves the following stages:

- establishment of object selection criteria that will satisfy user requirements;

- determination of the relations between objects and groups of objects;

- calculation of relations between objects in these interrelationships;

- determination of the method used to implement a specific selection process in accordance with established criteria based on the defined relationships and quantitative relations.

The apparatus of relation theory [5] was selected to establish the relationships between the objects. The following relations are used:

- $\varphi(x,y)$ is the objective relation defining the inclusion of object y on the list when object x is present;

- $\sigma(x, y), \bar{\sigma}(x, y)$ is the compatibility or incompatibility relation between x and y, respectively;

- $\rho(x, y)$ is the unconditional preference relation, i.e., object x is preferred over object y;

- $\rho_a(x, y)$ is the conditional preference relation, i.e., object x is preferred over object y "from the viewpoint" of a certain third object a called the "reference" object.

In order to eliminate contradictions or situations where contradictory elements appear within a set of objects when there is a large number of objects of various types certain weights of the objects are introduced in the relations. Two groups of weights are identified, the initial object weights (requirements) designated by the user and the expert-designated weights of the objects in the relations (for example, the expert would include a specialist in computer technology).

As the number of objects increases the process of expert weighting of the relations becomes much more complex. It is impossible in practice to represent quantitative weights when there are several thousand different objects. Only three object weight gradations (in descending order) were therefore used: necessary, desirable, and desirable yet not obligatory.

The selection of objects satisfying user requirements and set generation both require several stages. A certain set of objects is defined based on the objective relation for the given requirements. This set is the excess set since the relations between σ and σ were not accounted for in generating the set. The contradictions between the objects are then resolved. If we have the relation $\bar{\sigma}(x, y)$ and the weight of x exceeds the weight of y, object y is eliminated from the set. Personal computer models and their operating systems are used to generate the sets while the sets are presented to the user in accordance with the existing preference relations which are established between object groups M, B, and S.

All object groups are divided into three levels:

1. F, T - requirements;
2. A, P, C - subgoals;
3. M, B, S - goals.

The $\varphi(x, y)$ relations are established between objects on the first and second and the second and third levels, while the $\sigma, \bar{\sigma}$ relations are established between objects on the first and second levels and between objects on third level; the ρ and ρ_a relations are established between objects on the third level. The relational notation is as follows:

the φ relation:

$x : y_1^*, y_2^*, ..., y_k^*,$

where * is the weight of object y_i in the relation;

the $\sigma, \bar{\sigma}, \rho$ relations:

$x : y_1, y_2, ..., y_k;$

the ρ_a relation:

$x/a : y_1, y_2, \ldots, y_k.$

Such forms of notation indicate the existence of k relations between x and y_1, $y_2, \ldots, y_k$.

In defining the σ or $\bar{\sigma}$ relations the user or expert will select the relation whose indication requires the fewest actions.

The expanded object selection algorithm based on user requirements consists of the following:

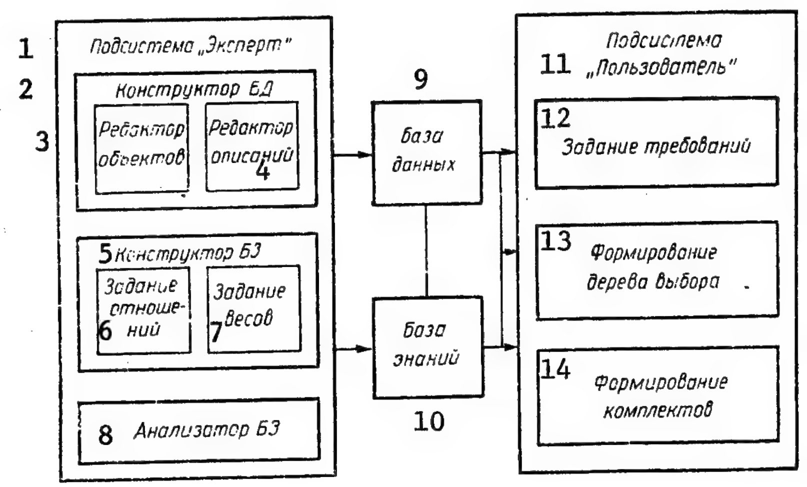


Figure 1. Layout of Personal Computer Hardware and Softwear Selection and Configuration Automation System

Key:

1. "Ekspert" subsystem
2. Data base designer
3. Object editor
4. Description editor
5. Knowledge base designer
6. Relational representation
7. Weight representation
8. Data base analyzer
9. Data base
10. Knowledge base
11. "User" subsystem
12. Requirement representation
13. Generation of selection tree
14. System generation

1. A selection tree is generated based on the relation $\wp$ to satisfy user requirements. The tree is a binary tree with the left indicators establishing a relation between the requirement levels and the right indicators establishing a relation between objects on the same level. Each tree element is assigned an expert-selected weight and a list of compatible (incompatible) objects.

2. A cycle is then run through the tree with a new weight assigned to each element. The weights are assigned on the principle of the highest weight of the lowest level element adopted by the lowest weight of the higher level element.

3. Contradictions between tree elements are resolved by one additional run by eliminating subtrees from the tree whose root has the lowest weight of the contradictory elements. When the weights are equal a request is sent to the user to review the weights of the user requirements causing such a contradiction.

4. Contradictory leaves are eliminated from the tree.

5. Sets of modules, blocks, and personal computer software components are configured by constructing integration trees whose first level will contain the personal computer model, the second level will contain the operating systems, the third level will contain other software components, and the fourth level will contain the support hardware or the additional automated workstation hardware.

6. The sets are ordered on the basis of preference relations.

A sample design of a personal computer hardware and software selection and integration automation system is shown in Figure 1.

The data base (BD) contains objects (requirements, subgoals, goals) and brief descriptions of the goal-objects, i.e., the objects from groups M, B, S.

The knowledge base (BZ) includes the relations between objects assigned weights. An information relation exists between the knowledge base and the data base.

The "Expert" subsystem supports interaction with the expert which is used to generate, supplement, and modify the data bases and knowledge bases. The object editor makes it possible to include, delete and edit object names in the data bases, and to relate these to the corresponding descriptions. The description editor is designed to generate the target object descriptions.

The knowledge base analyzer verifies the correctness of expert-defined relations. The following are verified:

- object compatibility (incompatibility) relations;

- paths in the selection tree that do not lead to the object leaves of groups M, B, S;

- consistency of preference relations given their transitive nature.

The requirements are established by the user by selection from a list provided by the system. The requirement weights are set up simultaneously. The user can correct the list as well as the requirement weights after selection.

The selection and integration trees are generated on the basis of the algorithm described above.

Expert-user interaction with the system which is based on menu-oriented and multiple-window interfaces includes a context-sensitive output of reference information. Figure 2 provides a sample screen image during interaction with the expert.

The system is based on MS DOS personal computers employing Pascal. The memory occupied by the software systems totals approximately 90 Kbytes. The data base contains of the order of 80 requirements in groups F and T, approximately 150 names of both domestic and foreign computer models, over 200 unit names, and over 300 software components (including primary programs only), and is constantly expanding. The number of units and peripherals includes both standard units (displays, printers, etc.) as well as professional personal computer modules (digital and analog input-output modules, CAMAC equipment interfaces, etc.) whose connection to personal computers yields automated workstations for various uses. The user requires at least 512 Kbytes of RAM and a 10 Mbyte hard disk.

The system can also be used as a reference system if the user indicates the subgoals or goals as requirements. The data and knowledge bases can be expanded by the user independently as information on new hardware and software becomes available.

"Personal computer" data base

Object groups	Personal computer models
Personal computer models	YeS 1840
Personal computer units	YeS 1841
Software	Neyron I9.66
	Neyron I9.69
	Iskra 1030
	Iskra 1130
	Iskra
	Mikro-16
Name of connected object	
YeS 1842	
	Pravets-16
	IZOT 1036 S
	IZOT 1037 S
	SM-1910-2

Figure 2. Sample screen image during the interactive mode with the expert.

REFERENCES

1. Nesterenko, N. V., Yakovlev, Yu. S. "Domestic personal computers and their application for workstation automation", Avtomatika, 1988, No. 5.

2. Yakovlev, Yu. S., Nesterenko, N. V. "One approach to developing professional personal computers", USiM, 1986, No. 2.

3. Yakovlev, Yu. S., Nesterenko, N. V., Zubenko, A. I., "Automated workstation for design and personal computer configuration selection", Personalnyye kompyutery i lokalnyye seti, Tbilisi, 1986.

4. Yakovlev, Yu., S., Nesterneko, N. V., "O probleme sozdaniya sovmestimogo ryada personalnykh EVM" [The problem of developing a compatible series of personal computers], Kiev, 1985 (Preprint AN USSR, In-t kibernetiki im. V. M. Glushkova, 85-56).

5. Rozen, V. V. "Tsel' - optimal'nost' - reshenie" [Goal - optimility - solution] Moscow, 1982.

UDC 681.3.015

Statistical Modeling on Personal Computers in the Interactive Mode

907G0042E Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 1989 pp 56-57

[Article by by D. N. Franchuk, N. V. Antonenko, and L. L. Snizhko]

[Text] Statistical modeling of complex stochastic processes has become widespread as a result of the development of fast, powerful computers. However further development has been restrained due to imperfections in the methods used to represent modeling results.

This situation can be solved by using interaction to formulate a mathematical model of the changing states of the system over time.

Statistical modeling of actual processes in the interactive mode involves the following steps.

Assume there is a complex stochastic process M characterized by the set of structural parameters $P_s(s = \overline{1, 7})$ varying over the range P_s min less than or equal to P_s less than or equal to P_s P_s min and forming a certain set of possible states of process M at time $t_j(j = \overline{1, c})$.

It is possible to formulate a mathematical mapping for process M as a statistical model W described by the set of information parameters $\pi_\psi(\psi = \overline{1, n}; n \leqslant N)$. The model W satisfies the requirements for adequate description of process M with deviation Δ which, in addition to the number of parameters of the model $\pi_\psi(\psi = \overline{1, n}; n \leqslant N)$ is inversely related to its effectiveness.

Any change in the number of information parameters $\pi_\psi(\psi = \overline{1, n}; n \leqslant N)$ will involve losses of adequacy of model W for describing process M.

The following sequence of calculations must be used in order to implement the statistical modeling process on a personal computer in the interactive mode:

- the entire set of information parameters $\pi_\psi(\psi = \overline{1, n}; n \leqslant N)$ of model W is divided into three groups:

1) the parameters whose range coincides with that of the parameters of the real process M;

2) the parameters whose values are selected by the expert during interaction with the personal computer since their range under specific conditions is substantially less than that of the parameters of abstract model W;

3) the parameters that can be taken as constant during a specific time period $t_j (j = \overline{1, c})$ for actual process M;

- an algorithm is generated for modeling process M; this algorithm requests the values of the parameters in the third group and requires expert input of the values of the parameters in the second group.

If the expert estimates of the parameters in the second group have a significant spread, the number of cycles in the calculation process grows by a factor of $k(k = \overline{1, l})$. The correctness of the decisions made by the expert during interaction with the personal computer is verified by the final modeling results.

We will consider a sample algorithm design used to generate a model for spare parts supply and demand for automobile enterprises. Utilization of the personal computer interactive mode for the modeling process makes it possible to substantially accelerate selection of the optimum solution. One feature of the interaction configuration is the implementation of the two request types:

- requests for the quantitative values of the arithmetic variables or their range (the second group of parameters $\pi_\psi(\psi = \overline{1, n}; n \leqslant N)$. Here the user inputs the numerical values of the quantities;

- requests for qualitative values of the characteristics [the third group of parameters $\pi_\psi(\psi = \overline{1, n}; n \leqslant N)$]. The user will respond with either a "yes" or a "no" to these requests.

The values of the following parameters are input in the initial modeling phase: characteristics of the incentive system F; the modeling period $t, t = \overline{1, 5}$; the spare part name $i, i = \overline{1, Q}$; and the factors $x = \{x_m\}$, $m = \overline{1, 7}$ characterizing the state of the local system G; and the annual quarter V; these are used to determine the demand for spare parts of the i^{th} name for the t^{th} period of the V^{th} annual quarter Y_{it}:

$$Y_{it} = [f(\{x_m\}) + 2\sigma]K_F,$$

where $f(\{x_m\})$ is the function describing the dependence of Y_{it} on the set of factors $x = \{x_m\}$, $m = \overline{1, 7}$; K_F is the coefficient accounting for the incentive system.

The array of values of the following parameters are input for subsequent calculations: Z'_{vi} - parameter characterizing the quantity of type i components; Z_{vi} - the characteristic indicating the presence of rebuilt spare parts of type i at the warehouse; R_{zi} represents the total inventory of the

i-type components; D_{zi} is the storage time of the i-type spare parts; O_{zi} is the factor determined by the ratio of the cost of rebuilt i-parts to the cost of new parts.

The following are then calculated on the local system level G:

- the level of satisfactory demand for i-type spare parts for the t period accounting for warehouse supplies U_{zit}

$U_{zit} = f(Y_{it}, R_{zi}, O_{zi})$;

- the volume of unsatisfied demand N_{it}

$$N_{it} = \begin{cases} 0 & \text{for} R_{zi} \geq Y_{it}, \\ Y_{it} - R_{zi} & \text{for} R_{zi} < Y_{it}; \end{cases}$$

- the excess supplies I_{it}

$$I_{it} = \begin{cases} R_{zi} - U_{zit} & \text{for} R_{zi} > U_{zit}, \\ 0 & \text{for} R_{zi} \leq U_{zit}; \end{cases}$$

- the cost of supplies A_{zit}

$A_{zit} = f(U_{zit}, I_{it}, N_{it}, D_{zi}, S_i, O_{zi})$

where S_i is the cost of the new i-type spare part.

If N_{it} not equal to 0 the calculations continue. If a requirement arises the delivery time of the i-type spare parts E_i and the size of the current lot of i-type spare parts R_{pi} are input. Then the level of satisfaction of the demand for i-type spare parts during the t period is determined accounting for delivery $U_{pit} = f(N_{it}, E_i, R_{pi})$ and the cost of delivery $A_{pit} = f(U_{pit}, S_i, T_{pi})$ where T_{pi} is the storage period of delivered - i - type spare parts.

Then the anticipated delivery volumes M_{qi}, delivery periods C_{qi}, batch assignment P_{qi} are indicated for the i-type spare parts together with the possibility of manufacture U_{qi} and the cost of manufacture O_{qi} after which the level of satisfaction of demand for i-type spare parts by manufacturing $U_{qit} = f(N_{it}, M_{qi}, C_{qi}, P_{qt})$ and the outlay required for this $A_{qit} = f(U_{qit}, S_i, O_{qi})$ are calculated.

In order to determine the change in demand accounting for the possibility for rebuilding the i-type parts or components U_{vi} the cost of this process S_{vi} is input together with the volumes of possible part restoration or component restoration O_{vi}, the restoration periods C_{vi}, the batch assignment P_{vi}, the percentage of restored i-type products or components B_{vi} and the percentage of the restored part or component supply R_{vi}. The possible restoration volume

of the i-type parts or components are determined during this period: $U_{vit} = f(N_{it}, O_{vi}, C_{vi}, P_{vi}, R_{vi}, B_{vi})$ together with the costs of restoration $A_{vit} = f(U_{vit}, S_{vi}, T_{iv})$, where T_{iv} is the storage period of the i-type restored parts or components.

This completes the statistical modeling process. The effective maximum satisfaction of demand N_{it} for the i-type spare parts with minimum outlay are then finally determined:

$$A_i = \min \{A_{pit} + A_{qit} + A_{vit}\}.$$

Utilization of the interactive process for this routine makes it possible to reduce substantially the duration of calculations when negative responses are obtained to key questions of parameter values (the third group parameters $\pi_\psi(\psi = \overline{1, n}; n \leqslant N)$.

The primary advantages of modeling in the interactive mode include:

- use of personal computers for solving many problems that were previously resolved solely by using supercomputers;

- a specific result can be obtained without the model losing its validity in describing an actual process within a time period that is acceptable for practical applications;

- improved effectiveness of optimum solution search by eliminating the requirement to simulate the entire possible range of parameters in the second group;

- the possibility for constructing self-training human-computer decision making systems that will have a significant effect on industrial management efficiency.

The supply and demand dynamics for spare parts (the ATP-23237, Irpen, Kiev oblast) was simulated in an interactive mode on the "Iskra-1030" personal computer; this configuration was also used to select efficient forms of labor organization for repair operations that are adequate for internal manufacturing processes (ATP-5 of the "Kievstroytrans" production association). The annual savings for these processes are 52 and 116 rubles per automobile, respectively.

UDC 681.3.06

Tools for Graphics Interfaces With Data Bases

907G0042F Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 1989 pp 58-59

[Article by V. V. Karpenko, V. V. Kolinko, A. N. Mendelev, and S. V. Shevchenko]

[Text] The interest in computer graphics as a means of establishing a user interface for computer operations has grown significantly in recent years. This can be attributed to the fact that, first, in many cases information representation in graphical form is substantially easier and natural for the human operator and, second, personal computers with sufficiently well-developed hardware and software tools for supporting computer graphics are widely available. However, using graphics software solely to support a user interface does not exhaust all potential capabilities for the use of video information. Specifically, an image which is a graphical representation of real world objects contains information on the spatial properties of such objects as well as their interrelationships, which can be used in decision making in a variety of subject domains.

The VISICOM tool package for creating graphics interfaces with data bases is a software system designed to input, store, retrieve, generate, and output for display graphics and alphanumeric descriptions of objects stored in a videographic data base. The need for such information arises in subject domains in which information on spatial characteristics is required in addition to alphanumeric descriptions of objects for decision making purposes. Such information is provided to the system user on the display screen as a set of geometrical samples of the necessary objects: various maps, sketches, transport and engineering facilities, etc. It is assumed that all geometrical samples of the objects are represented in a certain fixed coordinate system which we shall henceforth refer to as the graphical plane.

The VISICOM system is oriented for operation with the following types of two-dimensional geographical objects:

- linear objects characterized by linear extension and regular width (streets, engineering facilities);

- point objects characterized by position on the graphical plane and pictogram representation (theaters, shops, etc.);

- contour objects which have a border configuration (for example, country boundaries on a map);

- area objects: bordered objects with internal shading (forests, lakes, etc.).

All objects within the system are divided into classes such as parks, lakes, rivers, streets, etc. which in turn can be divided into subclasses. System users can define new object classes in accordance with their specific applications. In order to support utilization of the object classes each such class is assigned a pictogram which is then used in the request language to access the elements in this class. Each class is characterized by an expanded set of elements: the objects of this class. In addition to its graphical representation each object may also have an alphanumeric description and one or several visual representations.

VISICOM makes it possible to output to the display screen geometrical forms of all objects or a random set of objects from a selected set of classes. A pictogram menu is used to indicate the classes; this makes it possible to select one or several classes of objects for display or to inhibit certain classes of objects from appearing on screen. If the entire class is inhibited, no objects in this class will be displayed. Using the menu of objects it is possible to permit or inhibit the display of specific objects of any class permitted for display in an analogous manner.

The VISICOM system can be used for image scaling, to generate a scale border on screen whose dimensions and position can be manipulated by the user. According to user commands, only those objects appearing in the initial image within the scaled frame will be displayed on a magnified scale. The scaling operation can then be applied to the resulting image, etc.

When operating in the object menu mode the user can obtain on screen all text data on the present object as well as associated visual representations of this object. The user can also select one or several objects in order to obtain information on their spatial location. The selected objects blink on and off. It is then possible to obtain a response to the question of object position in the graphics plane.

The system can operate in a reference mode in which the display screen switches the cursor to an arrow. By moving the cursor the user can select any object displayed in the image. After guiding the cursor to the required position the information on the object indicated by the cursor will appear on screen. If the cursor indicates several objects simultaneously (i.e., a point belonging to several objects is indicated by the cursor) information on all such objects will appear sequentially on the screen.

The system data are stored in a text file prepared by the text editor in accordance with their description language. The geometrical representations

of the objects are stored in vectorial form which makes it possible, on the one hand, to minimize the occupied memory and, on the other, makes it possible to run a scaling operation. In addition to the data file the system uses a class description file which contains information on the procedures for outputting the geometrical dimensions of different classes to the screen as well as object color; a correlation is also established between the classes and associated pictograms. The pictograms are stored in a special file and are generated by means of the graphics editor.

The present version of the system has a number of limits:

- the system data base (DB) is a text file which precludes flexible data accessing;

- the alphanumeric data on the objects are stored as unstructured text which makes it impossible to use these data for multiaspect information retrieval;

- in the request mode there are no tools for representing the spatial interrelationships between the desired objects;

- the request language in the system is based on menu principles which makes it impossible to formulate complex retrieval requests for the data base.

Efforts at system enhancement are primarily oriented at eliminating these limitations in order to establish a videographic data base management system (DBMS) that will permit both storage and retrieval of structured alphanumeric information and graphics data.

The videographic information system data can be divided into two parts: alphanumeric and spatial (graphics) information. There are many commercial data base management systems available today that provide extensive capabilities of manipulating structured alphanumeric information. As far as spatial data bases are concerned, the corresponding software is essentially unavailable. This fact is also responsible for the need to develop a spatial data management subsystem in the course of development of the videographic data base management system; this would include development of logic and physical structures, methods of generating the spatial index, etc., which will make possible efficient storage of spatial data as well as rapid data access.

It is advisable to connect the graphics data management subsystem to one of the standard data base management systems to expand the capabilities of the latter for using such data in the process of supporting joint processing of alphanumeric and graphics data.

It should be noted that the relational algebra or relational calculus underlying the majority of request languages used in modern data base management systems are not suitable for use with graphics data, since operations and relations that are not found in traditional relational languages are characteristic of such data. The intersection of broken lines or the relation indicating the assignment of one region to another, etc.

represent examples of this. This makes it necessary to expand the relational algebra by including specific operations and relations used for spatial data processing. We assume that the request languages in the videographic data base management system will be based on such algebras, which have recently come to be called georelational.

In addition to these problems the significant amount of labor required to input the graphics data has made the issue of automating data base development an important issue. Such large volumes of data can be input only by using special preparation systems (image digitizers or scanners). Hence the videographic data base management system must have advanced data input and correction tools.

The system runs on the IBM PC XT class and the CGA color graphics equipment with a floppy disk drive, at least a 10 megabyte hard disk, a dot matrix printer, and a "mouse" manipulator. The use of IBM PC AT personal computers with an EGA monitor can substantially improve both the temporal characteristics of system operation and the quality of the resulting on-screen image.

UDC 681.3.06

Experience from Using PC-Based Information Technology at the 'Ukragropromremmash' Production Association

907G0042G Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 1989 pp 60-61

[Article by V. N. Timchenko and V. Z. Lekh]

[Text] The experience from using an automated enterprise management system (ASUP) at the Kiev "Agromash" factory - the base facility for incorporating "ASUP-repair facility" designs - confirms the general opinion that manufacturing automation should not begin with solving complex problems in order to achieve a maximum effect as quickly as possible. It is necessary to begin with automating the workstation where information is generated and processed. For example, at "Agromash" statistical data processing was first automated at the preparation shops and the sales department. This made it possible to obtain reliable information on the technical conditions and volume of the repair stock, on the dynamics of its arrival and the satisfaction of the product production plan as well as to prepare an information base for planning purposes.

Based on the "Agromash" experience the "Ukragropromremmash" which Production Association, includes this factory, began to implement in 1989 a comprehensive program to establish the ASU-REMMASH unified data processing service system. The designers had the following goals: to manage and plan all production operations based on information technology using personal computer computers; to establish a commercial information center to serve all factories and supply facilities of the association, to resolve local ASUP problems and incorporate paperless technology in management.

The incorporation of ASU-REMMASH represents one element in the technical retooling program of repair and machine construction factories in the Association and also represents a qualitatively new stage in its computerization. The development of the system is consistent with the primary directions of plans to develop a unified agroindustrial complex data processing system (YeS IVO APK) [1]. This data processing system of the Association includes all elements of personal computer-based information technology: word processing, spreadsheet calculations, personal data base (DB) management, expert system development, telecommunications, and electronic mail.

The increase in the volume of processed information has played an appreciable role in confirming the need to develop a comprehensive program as has the reduction in the size of the management staffs at all levels of the State Agroindustrial Committee, including the factories. Changes in the national economic management system have also contributed, specifically: delegation of the functions of central planning and coordination organs to the enterprise level, a conversion from a stock distribution of material resources to wholesale trading of resources and enterprise planning based on direct agreements.

The figure shows the information flow scheme before (a) and after (b) introduction of the ASU-REMMASH system.

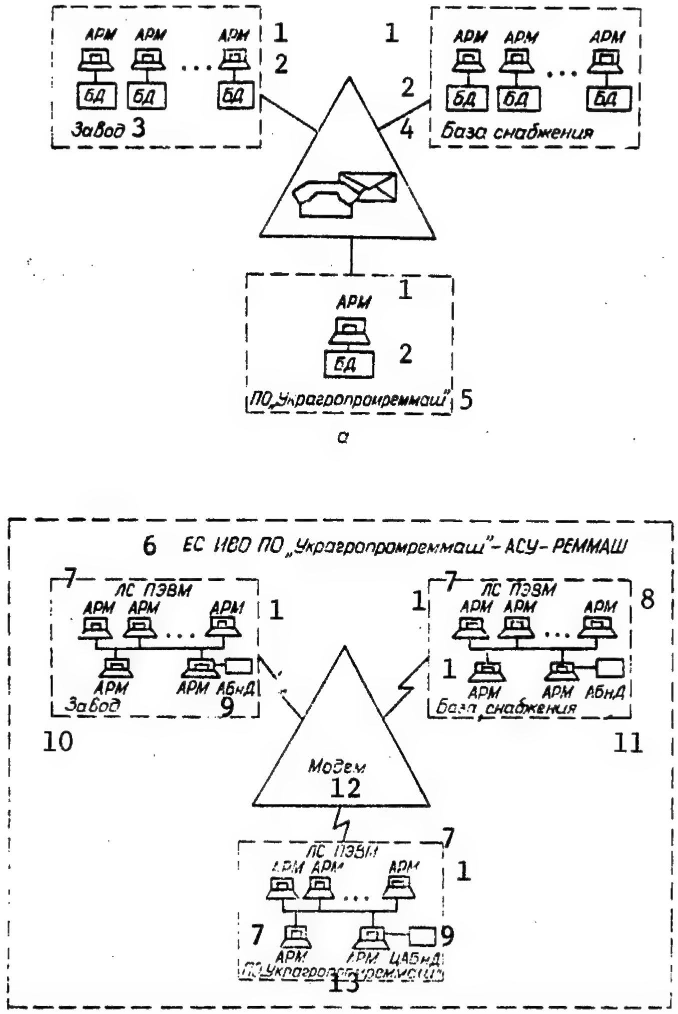
The integration of automated workstations that previously functioned independently into a single local-area network (LAN) at each facility (factory, supply base, Association management staff) is one of the requirements of personal computer-based information technology and is the primary organizational principle behind the structural design of the ASU-REMMASH system.

The central element in the "Ukragropromremmash" data processing system is the local-area network which consists of automated workstations installed for the Association staff and an automated data bank (ADB) containing information on the 42 factories and three supply bases of the Association. The information is accessed by direct data exchange between the factories or through the ASU-REMMASH centralized automated data bank. One of the purposes of the data bank is to generate long-range statistics for forecasting and future planning of the Association. Each data bank user can access only a specific portion of the information and is limited in the selection of tools for manipulating this data. For example, the Association managers and leading specialists have access to all the information at the same time that a feedback loop permits plants only partial access to data in the data base.

The primary characteristics of the local area network used at each facility are as follows: topology - trunk topology; physical environment - coaxial cable (75 ohms); accessing method - KDKN/OS [2-4].

From the methodological viewpoint centralized management of the development of the ASU-REMMASH system guarantees standard designs and proper maintenance of development and introduction sequences, and elimination of inefficient utilization of material and monetary resources as well as duplication of designs.

Standard ASUP design is based on state-of-the-art experience at repair and machine construction factories. Therefore the incorporation of standard ASUP designs at the "Ukragropromremmash" Production Association will improve the professional level of engineers and technicians. The automated knowledge bases which accumulate the experience of leading specialists within the field and scientists comprises the material base of future expert systems.



Information flow scheme at the "Ukragropromremmash" Production Association before (a) and after (b) incorporation of the ASU-REMMASH unified data processing system.

Key:

1. Automated workstation
2. Data base
3. Factory
4. Supply base
5. "Ukragropromremmash" production association
6. Unified agroindustrial data processing system at "Ukragropromremmash": the ASU-REMMASH system
7. Personal computer local area network
8. Automated workstation
9. Automated data bank
10. Factory
11. Supply base
12. Modem
13. "Ukragropromremmash" production association

The first stage of development of the ASU REMMASH system has used the results from previous solutions that have been tested on the base facility, the "Agromash" factory, and have also been used at the first Kiev automobile repair factory, the Shakhrikhan repair factory, the Lenin and Ak-Yarsk automobile repair factories of the Ukrainian SSR, at the computer center of the Ternopolskiy oblast agroindustrial committee of the Ukrainian SSR and the computer center of the Ulyanov oblast agroindustrial committee of the RFSFR. This stage will be completed when the first series of the ASU-REMMASH system is placed on line. The implementation period for the first series is 1989-1992.

The second series will be developed from 1993-1995 and will be completed after the acceptance of the standard design documentation, including all corrections.

The composition of the tasks in the third series will be determined during the design and introduction of the preceding series. The final version is planned for 1995 as a comprehensive program for the development of ASU-REMMASH through the year 2000.

The sequence for task introduction will depend on the information links. The management level tasks of the same level will be implemented simultaneously at all factories within a single quarter. As a rule the development of automated workstations for specialists, will have precedence over completion of automated data bank development, which will be used to integrate all automated workstations and support data exchange.

A total of 1.1 million rubles will be required for the design and development of the first and second series tasks in accordance with the comprehensive program for the development of the "Ukragropromremmash" unified agroindustrial complex data processing system. Moreover, capital investment for acquisition of computer equipment has already amounted to 1.5 million rubles.

The savings from the incorporation of the ASU-REMMASH system will be defined as the result of both technological and organizational innovations and will consist of the difference between total outlay for management prior to the incorporation of the ASU-REMMASH system and after this incorporation. The experience from using the automated enterprise management system at the "Agromash" factory suggests that the average annual savings for each factory will be 140,000 rubles for large factories, 70,000 rubles for medium-sized factories, and 60,000 rubles for small factories due primarily to savings in spare parts and materials (97.5 thousand rubles), improved work flow (from 0.91-0.95 to 0.96-1) and labor productivity (improved by 1.45%). The figures indicated in parentheses are actual data from the "Agromash" factory. Consequently the total savings from the introduction of the "ASU-REMMASH" system will run to approximately 3.2 million rubles for the entire Association.

REFERENCES

1. Yedinaya sistema informatsionno-vychislitelnogo obsluzhivaniya APK" [The APK unified data processing support system], Vorobev, I. A., Grzhibovskiy, S. P., Maskhuliya, et al., Moscow, 1988.

2. Ioffe, A. F., "Personalnyye EVM v organizatsionnom upravlenii" [Personal computers in organizational management], Moscow, 1988.

3. Madej, D., Marasek, K., Kurilowicz, K., Komputery Osobiste, Warsaw, 1988.

4. Nowakowcki, A., Mikrokomputery Buirowe, Warsaw, 1988.

UDC 621.376.52

Acousto-Optical Processors With Tunable Young Interferometer

907G0094A Novosibirsk AVTOMETRIYA in Russian No 6, Nov-Dec 1989 (manuscript received 6 Jan 89) pp 58-63

[Article by N.S. Vernigorov, A.S. Zadorin and A.V. Pugovkin (Tomsk) under the "Physical Aspects of Micro- and Optoelectronics" rubric]

[Text] **Introduction.** Implementation of functional capabilities of acousto-optical (AO) devices is related to a large degree to the problem of reading a two- or one-dimensional light field at the optical processor output. In practice, photodetectors (PD) are widely used for this purpose. The required accuracy of reading is achieved by providing the appropriate density and number of PD elements [1], but the possibility of their increase is limited, first of all by the available element base of photodetection equipment, and second of all by the complication of the PD electronic module.

Thus, a problem arises of improving accuracy characteristics of an AO processor (AOP) at a given density and number of sensitive PD elements.

Below, we examine a method for solving this problem. The essence of the method is that an improved accuracy of measurements is achieved by creating a fine structure of the detected light field, which (the structure) is generated during interference of two light beams, and by periodic time shifting of the interference pattern relative to the limited number of light sensitive elements. In this case, individual PD elements detect not discrete field intensity readings at their location points, but more detailed information on the profile of the interference pattern in the vicinity of those points, which is contained in photo current time response. It is the processing of these time signals that makes it possible to improve the accuracy of measurements.

To scan an interference pattern generated in diffracted light beams, a tunable phase shifter that changes phase difference between channels according to the following law:

$$\varphi = \Omega_0 t + \varphi_0. \qquad (1)$$

is added to one of interferometer channels.

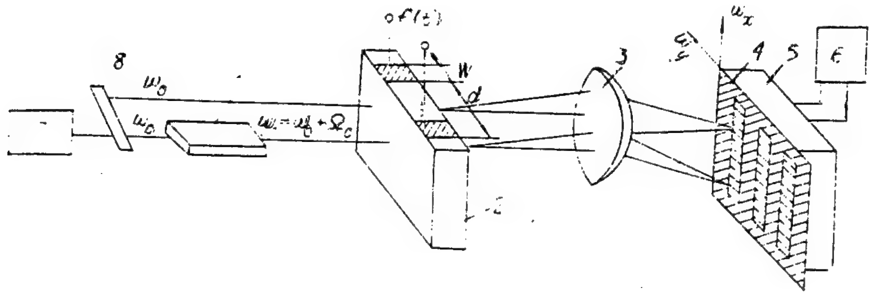


Figure 1

If interfering monochrome light beams are identical, then one can write the distribution of light field intensity in the focal plane of a lens with focus F in the following form [2]:

$$I(\omega_r, t) = F^2(\omega_r)\cos^2[(1/2)(\omega_r d + \Omega_0 t + \varphi_0)]. \qquad (2)$$

where $\omega_r = 2\pi r/\lambda F$ **is spatial frequency in the direction of coordinate r that connects centers of interfering light beams located at distance d from each other,** λ **is light wave length, and** $F(\omega_r)$ **is the three-dimensional spectrum of an individual light beam.**

It follows that linear time change φ **leads to a continuous shift of the interference pattern and harmonic oscillations (beats) of field intensity at point** ω_r **with frequency** Ω_0 **and initial phase that depends on spatial frequency** ω_r:

$$\psi = \varphi_0 + \omega_r d. \qquad (3)$$

For a given point $(\omega_r = \text{const})$ **the profile of the interference pattern in vicinity** $(\omega_r - (\pi/d)) - (\omega_r + (\pi/d))$ **will be contained in time dependence I(t). One can determine positions of maximums of function** $I(\omega_r)$ **by measuring phase** $(\psi - \varphi_0)$ **of electric signal beats at the photodetector output. Below, we shall examine several types of AOP with a tunable Young interferometer.**

1. Acousto-Optical Phase and Frequency Meter. A processor of this type is designed for panoramic measurement of frequency and phase difference φ_0 **of two radio signals.**

The functional diagram of an AOPF is shown in Figure 1; it is close to the one described in [3]. Here, the emission of monochrome source 7 is split into two light beams, whose frequencies are correspondingly shifted using device 1 by $\pm\Omega$, and then the beams are sent to two-channel AOM [acousto-optical modulators] 2. A non-stationary interference pattern is formed in the output plane of lens 3 and it illuminates photodetector line 5 via spatial filter 4. From the reference and signal photodetector elements, beat signals with frequency Ω_0 are sent to phase shifter 6.

There are different ways of performing the frequency shift operation in incident light beams. According to one method, we use an AOM operating in the Raman-Nath diffraction mode. When a monochrome signal with frequency Ω_M acts on the AOM, diffraction beams of the 1st and -1st order have a phase shift of $\pm\Omega_M$, respectively. In this case, $\Omega_0 = 2\Omega_M$.

At low frequencies Ω_0, it is more convenient to perform the operation of shifting the optical carrier frequency using electrooptical modulator EOM [4]. To do this, sawtooth control voltage is applied to the EOM; it shifts the phase of the light beam passing through the modulator according to a linear law in the 0-360° range with required shift frequency Ω_0.

The above described AOPF algorithm makes it possible to eliminate random optical phase shifts generated in interferometer channels, because the shifts will be introduced both to the measurement and reference channels. Besides, the AOPF has panoramic properties with respect to frequency. When the frequency of studied radio signals changes, diffracted light beams shift along coordinate ω_x, and beat signals are taken off other photodetector elements. When several pairs of radio signals with different frequencies act on the device input simultaneously, there are beat signals at outputs of different elements, wherein the photodetector element number determines signal frequency.

To describe the AOPF analytically, we shall use the equation of a generalized AO spectrum analyzer [5]. The response of a zero lag point photodetector in the case of monochrome input signals is described by the following expression:

$$e_{cp}(t) \sim |F(\omega_x)|^2 |F(\omega_y)|^2 \left[U_{1m}^2 + U_{2m}^2 + 2U_{1m}U_{2m}\cos(\Omega_0 t + \omega_y d + \varphi_0)\right], \tag{4}$$

where U_{1m} and U_{2m} are parameters that characterize amplitude non-identity of channels, and ω_x and ω_y are spatial frequencies along axes x and y, respectively, and in this case $\omega_x = \omega_y$.

For non-diffracted light beams, photodetector response is a reference signal relative to (4), derived as follows:

$$e_{cp.on}(t) \sim |F(\omega_x)|^2 |F(\omega_y)|^2 \left[E_{1m}^2 + E_{2m}^2 + 2E_{1m}E_{2m}\cos(\Omega_0 t + \omega_y d)\right], \tag{5}$$

where E_{1m} and E_{2m} are amplitudes of non-diffracted light waves in AOM channels.

It follows from (4) that, regardless of the frequency of input signals, photodetector response in an AO processor has the form of beats of frequency $\Omega_0 = \omega_1 - \omega_2$, which include the initial phase shift equal to φ_0.

By measuring phase difference of beat signals in the reference and information channels using a phase meter operating at fixed frequency Ω_0, we derive phase relationships in studied signals.

Because expressions (4) and (5) have been derived under the assumption of a point photodetector, we shall now examine the effect of its dimension along

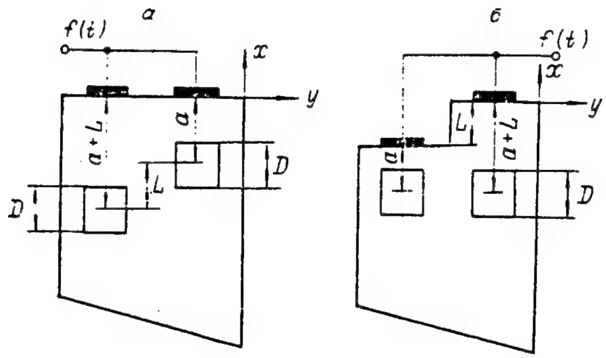


Figure 2

axis y. To do this, we shall integrate (1) along coordinate y within $y_{F0} \pm \delta y_F/2$, where y_{F0} is the coordinate of the photodetector center along axis y, and δy_F is the photodetector dimension along axis y.

Calculations demonstrate that at λ = 0.63 μm, F = 50 cm, d = 2 and W = 1 the photodetector dimension at which the beat amplitude is maximum is δy_F = = 78.75 μm, and the PD only detects an insignificant portion of the interference pattern energy.

We shall place mask 4 with a set of holes with period P, number of periods N and hole size $y' = \delta y_F$. in the output plane of lense 3 (see Figure 1). In this case, restrictions on the photodetector dimension along axis y are not as stringent. Assuming that function $F(\omega_y)$ changes slowly within δy_F we derive photodetector response as

$$e_{\text{cp}}(t) \sim \sum_{n=0}^{N/2} \left\{ \frac{\sin \frac{\pi W (n+1)}{2d}}{\frac{\pi W (n-1)}{2d}} \right\}^2 \left[1 + \cos\left(\Omega_0 t + \frac{2\pi d}{\lambda F} y_{F0} + \Delta\varphi\right)\right]. \qquad (6)$$

where W is the AOM converter dimension along axis y.

Calculations according to (6) demonstrate that at the above parameters of the optical circuit of an AO interferometer the amplitude of the photodetector output signal increases by 5.6 dB compared to the signal of a single photodetector with dimension δy_F.

The accuracy of measurement of phase characteristics of the input signal mainly depends on the accuracy of measurement of the phase difference of photodetector beat signals with phase meter 6 (see Figure 1).

2. Acousto-Optical Interferometers/Frequency Meters (AOF). AO interferometers can be used for improving characteristics of AO detectors/frequency meters. To do this, we shall connect both inputs of the device (see Figure 1) to each other and introduce an acoustic delay line with length L into one of the interferometer channels. Examples of such line combined with an AOM are shown in Figure 2. An advantage of this frequency meter is the

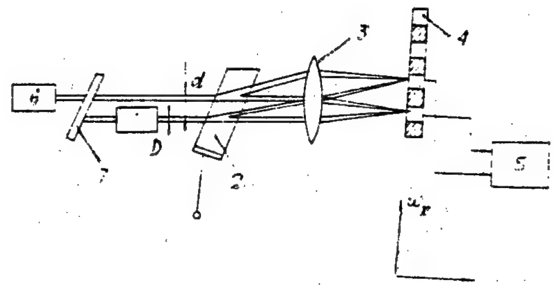


Figure 3

fact that it maintains frequency resolution of a one-channel AO device, while by measuring the phase difference between channels $\Delta\varphi = \Omega_x L/v$ it makes it possible to substantially improve the accuracy of carrier frequency determination. One can say that the frequency meter has two scales - a coarse scale for spatial frequency ω_x and fine scale for frequency ω_y. The accuracy of frequency measurement

$$\delta f_{min} = (v/2\pi L)\,\delta(\Delta\varphi) \qquad (7)$$

will be determined by delay time τ_0 and phase difference measurement error $\delta(\Delta\varphi)$. For frequency measurements on the coarse and fine scales to match, it is necessary that, when frequency changes by an amount equal to frequency resolution $\delta f_0 = v/D$ of the AO detector, phase difference changes by 2π. Therefore,

$$L = D. \qquad (8)$$

Another version of an AOF is a processor with a single-channel AOM and tunable intrerferometer, whose channels are located one under another in the diffraction plane (Figure 3). Components 1-7 in Figure 3 have the same function as in the AOP circuits discussed earlier. Interferometer tuning by means of frequency shifting device 1 results in interference pattern scanning along axis ω_x. i.e. along the coarse AOP scale. In this case, this coordinate is ω_r. Basic AOF characteristics (the fine scale, and the accuracy and unambiguity zone on the fine scale) can be evaluated based on expression (3), if it is assumed that $\varphi_0 = \Omega d/v$, $\omega_r d = -\Omega_{x0} d/v$, where $\Omega_{x0} = \omega_{x0}/v$ is the coordinate of the photodetector center adjusted to the angular time frequency, and the "-" sign is due to the Bragg character of diffraction [5]. Because $f_{x0} = \Omega_{x0}/2\pi$ determines a "coarse" frequency reading, frequency refinement is related to phase measurement $\psi = [2\pi(f - f_{x0})\,d]/v$ and the accurate correction is

$$\Delta f_T = f - f_{x0} = \psi v/2\pi d. \qquad (9)$$

The accuracy of frequency measurement δf_T on this scale depends on the accuracy of phase measurement $\delta\psi$:

$$\delta f_T = \delta\psi v/2\pi d. \qquad (10)$$

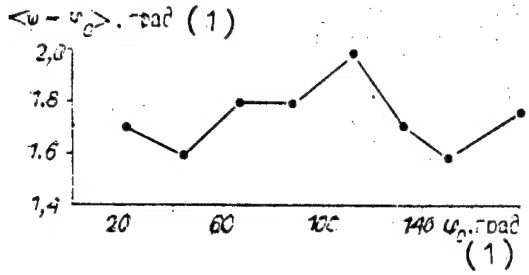


Figure 4

Key:

1. Degrees

Finally, the AOF zone of unambiguous measurement, i.e matching of the coarse and fine scales, are derived from condition $\psi_{max} = 2\pi$: $(\Delta f_r)_{max} = v/d$. Because the maximum value of a reading on the fine scale must be equal to AOF resolution on the coarse scale $\delta f_{rp} = v/D$. where D is the size of the optical aperture of each interferometer channel, we derive the following condition of matching the scales:

$$d = D. \qquad (11)$$

ψ is measured with phase meter 5, to whose inputs, as in the AOPF discussed earlier, one applies beat signals detected by PD elements illuminated by diffracted light beams and beats from main light beams detected by the PD reference element.

3. Experimental Results. 1. The AO phase and frequency meter described in section 2 [per original] was studied on a unit based on the diagram shown in Figure 1. Laser LG 52-2 was used as light source 7. The laser beam was split by plate 8 into two beams that were spread 2 mm apart and formed the interferometer arms. To tune the interferometer, an electrooptical phase shifter was placed in one of its arms. The shifter was made of a 24 x 8 x 2 mm lithium niobate plate, with sawtooth voltage applied to it in the direction of axis z. The sawtooth voltage generator was assembled according to circuit [6] and provided a 20 ns return time, 300 kHz frequency and 320 V voltage amplitude. This was sufficient for achieving φ_0 = 360°.

The two-channel AOM was also made of X-cut lithium niobate and excited by a longitudinal sound wave. Diffraction efficiency in the 200-400 MHz range was approximately 5% per 1 W of UHF power.

The dependence of relative beat phase difference ψ was measured experimentally in the reference and measurement channels on phase shift φ_0 between input signals, which was set using a coaxial phase shifter. As a PD, p-i-n photodiodes were connected to a videoamplifier with a 10 MHz frequency band. Phase difference ψ was measured using phase meter F2-16 with error $\delta\varphi \sim 5°$. Figure 4 shows the experimental

dependence of rms ψ' on φ_0. taken at a 370 MHz frequency. One can see that the deviation of ψ from φ_0 was within error $\delta\varphi$.

2. The study of an AO interferometer/frequency meter was conducted according to the circuit per Figure 2 in the short- and long-wave sections of the decimetric wave band. A beam from laser LG-52-2 was first collimated to size D = 6.3 mm. Then, the relative phase shift of two halves of the beam was changed according to a linear law using the above described electrooptical phase shifter with frequency Ω_0 = 300 kHz.

The AOF model was assembled from the same components as the AOPF model described earlier. For experiments at frequencies of approximately 2.5 GHz only a special AOM was used with the acoustic line made of Y-120°-cut lithium niobate excited by a slow shear acoustic wave that had high diffraction activity for a wideband abnormal AOV [not further identified]. The modulator was controlled by generator G3-10A, whose frequency was monitored by frequency meter Ch3-34A, and operated in the 2.1-2.6 kHz range with diffraction efficiency of approximately 3%.

The objective of the experiments was to verify frequency dependence (9) that determines the AOF fine scale. As a result of measurements of phase shift ψ as a function of $\Delta/_T$ taken at frequency 320 MHz and aperture 2.5 mm and at frequency 2.51 GHZ at aperture 1.5 mm, it has been found that the deviation of the above dependences from (9) did not exceed the error of phase shift measurement $\delta\varphi = 5°$.

BIBLIOGRAPHY

1. Kulakov, S.V., "Akustoopticheskiye ustroystva spektralnogo i korrelyatsionnogo analiza signalov" [Acousto-Optical Devices for Spectral and Correlation Analysis of Signals], Leningrad, "Nauka", 1978.

2. Born, M. and Wolf, E., "Osnovy optiki" [Fundamentals of Optics], Moscow, "Nauka", 1970.

3. Bokov, L.A., Krakovskiy, V.A. and Pugovkin, A.V., "Measurement of Mutual Spectral Power Density With Acousto-Optical Processors", RADIOTEKHNIKA I ELEKTRONIKA, 1982, Vol XXVIII, No 6.

4. Vernigorov, N.S., Zadorin, A.S. and Lukinskiy, S.V., "Akustoopticheskiy fazometr-chastotomer" [Acousto-Optical Phase and Frequency Meter], USSR Certificate of Authorship No 1334093, published in BI [Bulletin of Discoveries, Inventions and Trade Marks], 1987, No 32.

5. Pugovkin, A.V., "On Theory of Bragg Spectrum Analyzers", in book "Akustoopticheskiye metody i tekhnika obrabotki informatsii" [Acousto-Optical Methods and Information Processing Technology], Leningrad, LIAP, 1980, No 142.

6. Dyakonov, V.P., "Lavinnyye tranzistory i ikh primeneniye v impulsnykh ustroystvakh" [Avalanche Transistors and Their Application in Pulsed Devices], Moscow, "Sovetskoye radio", 1973.

UDC 681.333(088.8):535.345.6

Synthesis of Interference Optical Coatings

907G0094B Novosibirsk AVTOMETRIYA in Russian No 6, Nov-Dec 1989 (manuscript received 6 Apr 88) pp 63-69

[Article by Ye.G. Stolov (Leningrad) under the "Physical Aspects of Micro- and Optoelectronics" rubric]

[Text] One of the most important tasks of computer engineering is the development of new methods for solving complex nonlinear equations generated in the process of synthesis of multicomponent systems with specified physical properties.

Analysis of data accumulated to date makes it possible to conclude that searching for solutions on a net with subsequent optimization is one of the most promising directions for solving this problem, because this method is generally applicable, does not call for the zero approximation and may be used for all types of the quality functional.

To successfully implement this method, one must develop a theory that ensures the maximum possible speed when checking points belonging to the range of unknowns for correspondence to an equation.

To develop such theory, analysis was made of general properties of actual complex physical systems, on the one hand, and of the capabilities of modern computers, which are so far the only tool for solving cumbersome information processing problems, on the other.

Moreover, the specific character of the problem under consideration was taken into account: checking the maximum possible number of various physical systems for the presence of required properties in the shortest possible time.

An important feature of actual physical systems is the possibility of presenting them in the form of several simpler subsystems and describe concrete properties of the system as a whole using several "integral parameters" characterizing each subsystem. The term "integral parameters" is used in the sense that they fully characterize physical properties of a subsystem given certain interactions and do not contain specific information on the design configuration of each subsystem.

For instance, assume there is a system consisting of a large number of electric charges and one has to analyze the action of such system on a trial charge located in a certain space region. The problem can be solved by computing the field of each charge in the system and adding vectorially all these fields in order to derive the resulting field. Another method, that has formed the foundation of the developed theory, consists in splitting the entire system into several subsystems, determining the strength of the field generated by each subsystem and adding subsystem field strengths, and in this latter case the time spent on calculating one alternative is considerably shorter than in the former case. The strength of the electric field generated by a subsystem is an "integral parameter" that describes all its properties that are essential within the framework of the problem being solved, i.e. it characterizes the subsystem's interaction with other charges, while at the same time it does not contain data on the number, magnitude and location of charges comprising the subsystem.

[illegible] ... is the presence of such subsystems, which, when interacting with the initial system, do not significantly affect its physical properties. For instance, if one places a system of electric charges inside a uniformly charged sphere, the field will not change. This property was also used when developing the new synthesis theory.

On the other hand, properties of modern computers were taken into account - the capability to store large data arrays in RAM and rapidly extract needed numbers from the RAM when performing computations, as well as the fact that the same signal can be simultaneously sent to several decoders, each connected to its own memory module, and the capability of extracting from each module numbers needed for further computations. Moreover, consideration was given to the fact that, because the computation of values of even elementary functions takes up a lot of machine time, the number of such operations must be as low as possible.

We shall now take into account the possibility of presenting a complex system in the form of several subsystems. Let the initial system under consideration be split into C subsystems. Obviously, if for each subsystem we assume S versions, then, having in the RAM data on CS subsystems, one can analyze $M = S^C$ systems. For instance, at $C = 3$ and $S = 100$, $M = 10^6$, i.e. if a RAM contains previously stored information on a small number of subsystems included in a complex system, one can analyze properties of a large number of versions of the complex system.

We shall now derive the general form of a nonlinear equation that describes physical properties of a complex system in terms of "integral parameters" characterizing properties of subsystems. Let the i-th subsystem ($1 < i \leqslant C$)

be described by "design" parameters $X^{(i)}_{r_1}, X^{(i)}_{r_2}, \ldots, X^{(i)}_{r_{k_i}}$. For instance, in

the above discussed case of an electrical system of charges, charge coordinates and values are such parameters.

An "integral parameter" that describes the i-th system is a function of the above listed design parameters and can be written as the following function:

$\varphi_i\left(X^{(i)}_{r_1}, X^{(i)}_{r_2}, \ldots, X^{(i)}_{r_{k_i}}\right)$. Depending on the problem, there can be several such parameters. For instance, in the case of a system of electrical charges there are three such parameters - electric field strengths in directions x, y and z. The following expression describes properties of a complex system in terms of "integral parameters":

$$F\left(\varphi_1\left(X^{(1)}_{r_1}, X^{(1)}_{r_2}, \ldots, X^{(1)}_{r_{k_1}}\right), \ldots, \varphi_C\left(X^{(C)}_{r_1}, X^{(C)}_{r_2}, \ldots, X^{(C)}_{r_{k_C}}\right)\right). \qquad (1)$$

Then, the nonlinear equation that all $\{x_i\}$ must meet in order for a complex physical system to have the required set of properties takes the following form:

$$F\left(\varphi_1\left(X^{(1)}_{r_1}, X^{(1)}_{r_2}, \ldots, X^{(1)}_{r_{k_1}}\right), \ldots, \varphi_C\left(X^{(C)}_{r_1}, X^{(C)}_{r_2}, \ldots, X^{(C)}_{r_{k_C}}\right)\right) = A, \qquad (2)$$

i.e. it is a superposition of functions of unknowns. Note that the same unknowns can be included in various superposition functions.

Taking the above into account, this work proposes the following idea for solving equations of type (1), which makes it possible to implement the search on a uniform net in the space of unknowns, while ensuring the maximum possible speed. Values of functions φ_i, i = 1, 2, ..., C, for points in the range of unknowns at which the search is conducted are computed beforehand and entered into a computer RAM under certain numbers that depend on the set of values $\left(X^{(i)}_{r_1}, X^{(i)}_{r_2}, \ldots, X^{(i)}_{r_{k_C}}\right)$ where they are stored during the entire computation process. A separate memory module is formed for each φ_i, and the total number of such memory modules is equal to C. Each point in the range of unknowns that is checked for correspondence to equation (2) is assigned a certain number, depending on values of all unknowns at this point. During the computations, position codes of numbers of points in the range of unknowns are successively formed in a certain register. Information on numbers of cells in memory modules that store values of φ_i, i = 1, 2, ..., C, is taken from this register. According to these numbers, all C memory modules are addressed, and required values of functions φ_i are extracted from the modules. Then, function $F(\varphi_1, \varphi_2, \ldots, \varphi_C)$ is computed, and its value is compared to number A on the right-hand side of equation (2).

We shall now describe the analytical scheme of computations. Without any loss of generality, we shall examine the following equation:

$$F\left(\alpha\left(X_{k^{(1)}_1}, X_{k^{(1)}_2}, \ldots, X_{k^{(1)}_m}\right), \beta\left(X_{k^{(2)}_1}, X_{k^{(2)}_2}, \ldots, X_{k^{(2)}_m}\right), \gamma\left(X_{k^{(3)}_1}, X_{k^{(3)}_2}, \ldots, X_{k^{(3)}_m}\right)\right) = A. \qquad (3)$$

where X_i are unknowns, wherein i = 1, 2, ..., 2m, i.e. the equation has 2m unknowns and each superposition function has m unknowns. The same unknowns can be included in different superposition functions. We shall seek solutions on a uniform net in the space of unknowns, wherein each unknown can assume t

values. All in all, t^{2m} points will be analyzed as a result of computations. Point numbers change in the $0 \leqslant N < t^{2m}$ range. Point number N determines unambiguously values of unknowns $X_1, X_2, \ldots, X_{2m}$ in accordance with the following formulas:

$$X_i(N) = X_i + \frac{X_i^{(2)} - X_i^{(1)}}{t-1} a_i, \quad i = 1, 2, \ldots, 2m;$$
$$N = a_1 + a_2 t + \ldots + a_{2m} t^{2m-1}, \qquad (4)$$

where $[X_i^{(1)}, X_i^{(2)}]$ is the the range of X_i, $i = 1, 2, \ldots, 2m$; a_i is a non-negative integer, $a_i < t$. Thus, number $a_1, a_2, \ldots, a_{2m}$ represents number N in a t-th numbering system.

The process of solving equation of type (3) goes as follows (see the Figure).

1. The values of function

$$\alpha_l = \alpha\left(X_{k_1^{(1)}}\left(a_{k_1^{(1)}}\right), X_{k_2^{(1)}}\left(a_{k_2^{(1)}}\right), \ldots, X_{k_m^{(1)}}\left(a_{k_m^{(1)}}\right)\right), \qquad (5)$$

$$l = a_{k_1^{(1)}} + a_{k_2^{(1)}} t + \ldots + a_{k_m^{(1)}} t^{m-1}, \quad 0 \leqslant l < t^m; \qquad (6)$$

$$\beta_p = \beta\left(X_{k_1^{(2)}}\left(a_{k_1^{(2)}}\right), X_{k_2^{(2)}}\left(a_{k_2^{(2)}}\right), \ldots, X_{k_m^{(2)}}\left(a_{k_m^{(2)}}\right)\right), \qquad (7)$$

$$p = a_{k_1^{(2)}} + a_{k_2^{(2)}} t + \ldots + a_{k_m^{(2)}} t^{m-1}, \quad 0 \leqslant p < t^m; \qquad (8)$$

$$\gamma_q = \gamma\left(X_{k_1^{(3)}}\left(a_{k_1^{(3)}}\right), X_{k_2^{(3)}}\left(a_{k_2^{(3)}}\right), \ldots, X_{k_m^{(3)}}\left(a_{k_m^{(3)}}\right)\right), \qquad (9)$$

$$q = a_{k_1^{(3)}} + a_{k_2^{(3)}} t + \ldots + a_{k_m^{(3)}} t^{m-1}, \quad 0 \leqslant q < t^m. \qquad (10)$$

are calculated on a computer. Computed values of α_l, β_p and γ_q are retained in their memory modules under numbers l, p and q, respectively.

2. Values of N, from 0 to $(t^{2m} - 1)$, are searched linearly, and for each N respective values of l, p and q are derived from formulas (6), (8) and (10). From the derived l, p and q, values of $\alpha_l, \beta_p, \gamma_q$ are extracted from memory modules, and from these, values

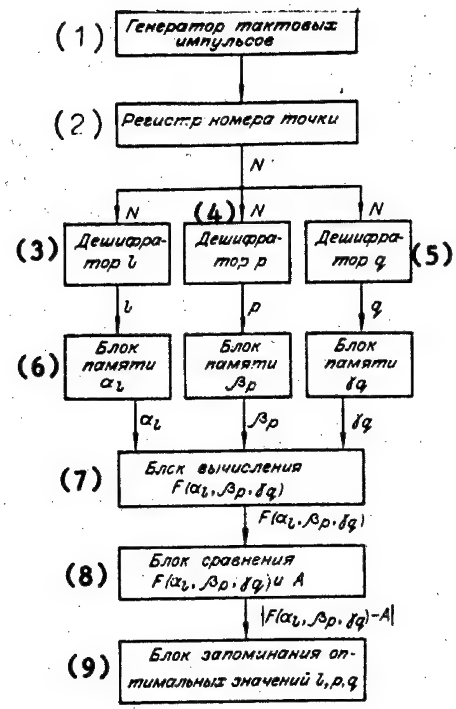
$$F(\alpha_l \beta_p \gamma_q), \quad 0 \leqslant l, p, q < t^m, \qquad (11)$$

i.e. values of the function on the left-hand side of equation (3) at the point that belongs to the range of unknowns - $X_1(a_1), X_2(a_2), \ldots, X_{2m}(a_{2m})$, are successively computed.

3. During the computations, values of l, p and q for which relationship

$$A - \varepsilon < F(\alpha_l, \beta_p, \gamma_q) < A + \varepsilon, \quad (12)$$

is satisfied are identified, i.e. one determines with accuracy ε points from



Key:

1. Clock pulse generator
2. Point number register
3. Decoder 1
4. Decoder p
5. Decoder q
6. Memory module
7. Computation module
8. Comparison module
9. Module for storing optimum values of 1, p and q

the range of unknowns that satisfy equation (3). The computer memory retains triplets of values of 1, p and q for which $|A - F(\alpha_l, \beta_p, \gamma_q|$ is minimal.

4. For these three values of 1, p and q, the corresponding values of $X_1, X_2, \ldots, X_{2m}$ are determined; 1, p and q are presented in the t-th numbering system according to formulas (6), (8) and (10), and then the values of $a_1, a_2, \ldots, a_{2m}$ obtained according to formula (4) determine the values of $X_1, X_2, \ldots, X_{2m}$.

5. Then, the computed values of unknowns are optimized, for instance, in accordance with the gradient descent program or using the relaxation method [1, 2].

The advantage of the described computation scheme is in achieving the maximum possible speed when searching for solutions on a net, due to complete prevention of duplication of computational operations when analyzing various

points in the range of unknowns. This effect is achieved because values of functions φ_i are not recomputed when checking each point in the range of unknowns, but taken from the computer RAM. The speed of the method depends on time spent on computing function $F(\alpha_i, \beta_p, \gamma_q)$ and, which is very important, it does not depend on the number of elements (for instance, interference coating layers) comprising a complex system. Besides, another advantage of the developed theory is its capability to take into account very complex interactions, when parameters of the same elements are included in descriptions of different subsystems.

We shall illustrate the developed algorithm with an example of solving the problem of synthesis of interference coatings with required spectral characteristics.

The objective of the computation is to determine optical thicknesses that at wavelengths of $\lambda_1, \lambda_2, \ldots, \lambda_r$ make it possible to obtain specified values of energy transmittance. Refractive indices of layers and media framing the coating are assumed to be fixed.

In a theoretical consideration, any interference coating can be presented in the form of a $\Pi_1 - P - \Pi_2$, structure, where P is the dividing layer, whose role can be played by any internal layer of a coating, and Π_1 and Π_2 are parts of coating to the left and to the right of the dividing layer. In the case of normal incidence of light flux and absence of absorption in layers, energy transmittance T corresponding to wavelength λ is derived from the following formula:

$$T = \frac{\left(1-\rho_1^2(\lambda)\right)\left(1-\rho_2^2(\lambda)\right)}{1+\rho_1^2(\lambda)\rho_2^2(\lambda)-2\rho_1(\lambda)\rho_2(\lambda)\cos\left(\Delta_1(\lambda)+\Delta_2(\lambda)+\frac{4\pi n_a d_a}{\lambda}\right)}. \qquad (13)$$

where $\rho_1(\lambda)$ and $\rho_2(\lambda)$ are modules of amplitude reflectivities of both parts of coating Π_1 and Π_2; $\Delta_1(\lambda)$ and $\Delta_2(\lambda)$ are phase shifts of an electrical vector reflected on the side of the dividing layer for coating parts Π_1 and Π_2 ; and n_a and d_a are the refractive index and geometric thickness of the dividing layer. A (2m + 1)-layer interference coating can be presented in the following form: Π_1 are the first m layers; P is the dividing (m + 1)-st layer, and Π_2 are the (m + 2)-, (m + 3)-, ..., (2m + + 1)-st layers.

In this case, in order to derive optical thicknesses of a coating with required energy transmittances $T_1, T_2, \ldots, T_r$ at wavelengths $\lambda_1, \lambda_2, \ldots, \lambda_r$, respectively, we have the following system of r equations:

$$\begin{gathered} T(\rho_1(X_1, X_2, \ldots, X_m, \lambda_f), \Delta_1(X_1, X_2, \ldots, X_m, \lambda_f), \\ \rho_2(X_{m+2}, X_{m+3}, \ldots, X_{2m+1}, \lambda_f), \Delta_2(X_{m+2}, X_{m+3}, \ldots \\ \ldots, X_{2m+1}, \lambda_f), X_{m+1}, \lambda_f) = T_f, \; f = 1, 2, \ldots, r, \end{gathered} \qquad (14)$$

where $X_1, X_2, \ldots, X_{2m+1}$ are optical thicknesses of coating layers.

Values of ρ_1, Δ_1, ρ_2 and Δ_2 are computed using the well known algorithm [4].

A program implementing the above described method for numerical solution of nonlinear equations as they apply to the problem of synthesis of interference optical coatings was compiled for YeS 1040 computers. The following source data were entered into a computer: $[X_i^{(1)}, X_i^{(2)}]$, $i = 1, 2, \ldots, (2m + 1)$, are boundaries of ranges of optical thicknesses of coating layers in units of $\lambda_0/4$ (λ_0 is a conventional unit of measurement of optical thicknesses of coating layers and emission wavelengths); integers t and r; $\lambda_1, \lambda_2, \ldots, \lambda_r$ in units of λ_0 and $T_1, T_2, \ldots, T_r$; $n_0, n_1, \ldots, n_{2m+1}, n_{2m+2}$ are refractive indices of coating layers and framing media. After performing computations according to the above program, coating design was optimized according to the gradient descent program. After computations were completed, optical thicknesses of layers before and after optimization were printed out.

Below, we present results of calculation for several types of coatings. Table 1 shows designs of anti-reflection coatings for three wavelengths for various values of substrate refractive indices n_{Π}. Refractive indices of layers are $n_B = 2.0$ and $n_H = 1.45$, which correspond to the pair ZrO_2-SiO_2, wherein SiO_2 layers are adjacent to the substrate and air.

Layer thicknesses are given in units of $\lambda_0/4$ and wavelengths are given in units of λ_0 Layers are numbered starting at air. Below, values of energy reflectivities of coatings for specified wavelengths are given. Column 1 shows values of optical thicknesses before optimization,and column 2, after optimization. The results confirm the effectiveness of the proposed computation scheme.

This approach to the synthesis of complex physical systems based on taking into account their properties also makes it possible to expand the capabilities of optimization methods [1 and 2]. A major drawback of most existing optimization methods is their local character, i.e. the fact that they make it possible to perform search in a narrow parameter range in the vicinity of a local minimum of the quality functional. Some authors propose to remove this limitation by adding additional subsystems to an already computed system and by subsequent optimization of parameters of the so formed system [5 and 6]. However, those authors give no criteria for choosing parameters of added subsystems, and if those parameters are chosen at random, system characteristics can deteriorate so much that subsequent optimization does not even make it possible to achieve the initial level.

It has been noted earlier that one of the most general features of most physical systems is the presence of subsystems which, when interacting with the initial primary system, do not change its properties in a specified space or spectrum zone or in the specified time range.

It is advisable to add such subsystems to a computed system in order to continue the optimization process, because the addition does not considerably deteriorate the system's properties, while at the same time it makes it possible to continue optimization, because the number of variable parameters

Table 1

(1) Длины волн	$\lambda_1=0,5\lambda_0$ $\lambda_2=1,0\lambda_0$ $\lambda_3=1,2\lambda_0$									
(2) Показатель преломления подложки	$n_П=1,52$		$n_П=1,60$		$n_П=1,65$		$n_П=1,70$		$n_П=1,75$	
(3) Номер слоя	1	2	1	2	1	2	1	2	1	2
1	1,400	1.388	1.400	1.396	1.400	1.401	1,200	1.205	1,200	1,200
2	0,600	0.597	0,600	0,589	0.600	0.588	1.200	1.187	1.200	1.200
3	1.200	1.158	1.200	1.188	1.200	1.194	0,200	0.212	0.200	0,200
4	0.200	0,194	0.200	0.185	0.200	0.184	1.400	1.391	1.400	1.400
5	1,400	1.409	1.400	1.396	1.400	1.386	0.200	0.193	0 200	0.200
6	0.200	0.190	0.200	0.215	0.200	0.229	1.000	0.986	1.000	1.000
7	1.000	1.014	1.000	1.008	1.000	1,017	0.200	0.193	0.200	0.200
R_1, %	0,01	0.10	0,10	0,05	0.15	0,05	0.10	0,05	0,05	0,05
R_2, %	0,35	0,25	0,20	0,05	0.10	0,05	0.10	0,05	0,05	0,05
R_3, %	0,20	0,20	0,05	0,05	0,10	0,05	0,01	0,01	0,05	0,05

Key:

1. **Wavelengths**
2. **Substrate refractive index**
3. **Layer number**

has increased. As in the previous case, we shall now illustrate this approach using the example of synthesis of interference coatings with specified optical properties. In this case, parameters of added layers can be chosen based on the theory of equivalent layers [7 and 8]. According to this theory, any symmetrical system of films, as far as its optical properties are concerned, is identical to one film with effective refractive index N_E and effective optical thickness Γ_E, wherein N_E and Γ_E vary depending on emission wavelength. It is well known that sections of type

$$(qL)[2(1-q)H](qL) \quad \text{or} \quad (qH)[2(1-q)L](qH), \tag{15}$$

where [2-(1-q)H] and [2(1-q)L] are layers with a high n_B and low n_H refractive index, respectively, and optical thickness $2(1-q)\lambda_0/4$; (qH) and (qL) are layers **with a high and low refractive index, respectively, and optical thickness** $q\lambda_0/4$ and **change their effective refractive index in the $n_н \leqslant N_E \leqslant n_в$ range when q varies in the [0, 1] range. And the above systems hardly change the value of N_E in a wide range of wavelengths. This fact makes it possible to use them as additional groups of layers introduced into a coating. Into a coating whose optical properties must be improved and to which regular optimization methods add no further improvement, one or several systems of type (15) are introduced between the interference coating and a framing medium. In doing this, the value of q is chosen such that the value of N_E is approximately equal to the refractive index of the framing medium. Then, values of optical thicknesses are optimized using the gradient or relaxation method. Then, the procedure can be repeated. For illustration purposes, we shall now examine calculation of an achromatic anti-reflection coating. A five-layer antireflection coating whose design is shown in Table 2, columns 1-3 was chosen as the initial design. During the computations, system**

Table 2

Номер слоя	Показатель преломления слоя	Оптические толщины слоев в единицах $\lambda_0/4$ (отсчет от подложки)			
1	2	3	4	5	6
1	1,46	1.775	0,029	0,042	0.047
2	2,20	0,221	0,022	0,035	0,044
3	1.46	0.221	1.928	0,103	0,074
4	2.20	1.569	0.217	0,022	0,007
5	1.46	0,852	0.225	2.017	0,087
6	2.20		1.593	0,253	0,018
7	1.46		0,854	0,189	2.030
8	2.20			1,057	0.303
9	1.46			0,858	0.164
10	2.20				1.414
11	1.46				0,871
$\Phi \cdot 10^3$		644	530	411	351

Key:

1. **Layer number**
2. **Layer refractive index**
3. **Optical thicknesses of layers in units of (counting from the substrate)**

(0.05L)(0.1H)(0.05L) was introduced three times between the substrate and coating, and each time the design was subsequently optimized using the gradient descent method. The quality functional had the following form:

$$\Phi(\mathbf{X}) = \int_{\alpha_1}^{\alpha_2} [R(\mathbf{X}, \alpha) - F(\alpha)]^2 \, d\alpha, \qquad (16)$$

where X is the set of variable parameters; $F(\alpha)$ **is the required spectral characteristic of the coating in the** $[\alpha_1, \alpha_2], \alpha = \frac{\pi}{2}\frac{\lambda_0}{\lambda}$ **range; and** λ **is emission wavelength in vacuum. Designs of coatings derived as the result of computations are shown in Table 2 (the last line shows values of the quality functional). One can see from the Table that as the number of layers increases, the value of** Φ **decreases, i.e. coating quality improves, which proves the effectiveness of the above examined scheme.**

Thus, the proposed approaches to the problem of synthesis, which take into account physical properties of actual systems, are promising. They are generally applicable. To adapt them to various problems, one only has to develop a mathematical description of properties of a complex system in terms of "integral parameters" of subsystems and a methodology for designing subsystems that have no significant effect on physical properties of the system they interact with.

BIBLIOGRAPHY

1. Baumeister, P.W., "Design of Multilayer Filters by Successive Approximations", JOSA, 1958, No 48 p 955.

2. Zycha, H., "Refining Algorithm for the Design of Multilayer Filters", APPL. OPT., 1973, No 2 p 979.

3. Furman, Sh.A., "Tonkosloynyye opticheskiye pokrytiya" [Thin-Layer Optical Coatings], Leningrad, "Mashinostroyeniye", 1977.

4. "Fizika tonkikh plyonok. T. 5" [Physics of Thin Films. Vol 5], G. Hass and R.E. Tun editors, Moscow, "Mir", 1972.

5. Shatilov, A.V. and Tyutikova, L.P., "Example of Computation of Interference Light Filter Using Successive Synthesis Method", OPTIKA I SPECTROSKOPIYA, 1963, No 14 p 426.

6. Pelletier, E., Klapisch, M. and Giacomo, P., "D'empilements de Couches Minces", NOUV. REV. d'OPT. APPLIQUEE, 1971, Vol 2, No 5 p 247.

7. Herpin, A., "Calcul du Pouvoir Reflecteur d'un Systeme Stratific Quelconque", COMP. REND. ACAD. SCI., 1947, No 182 p 225.

8. Epstein, L.I., "The Design of Optical Filters", JOSA, 1952, Vol 42, No 11 p 806.

UDC 535.241.13:537.228

Hybrid TSLMs Based on Electrooptic Ceramics-MOSFET Structures

907G0094C Novosibirsk AVTOMETRIYA in Russian No 6, Nov-Dec 1989 (manuscript received 9 Jan 89) pp 70-75

[Article by A.V. Guk, V.B. Zalesskiy, Ye.G. Paperno and V.A. Pilipovich (Minsk) under the "Physical Aspects of Micro- and Optoelectronics" rubric]

[Text] **Introduction. Optic printers designed for high-speed output of computer alphanumeric and graphics information have been developed as an alternative to mechanical printers, whose parameters, and particularly speed, do not anymore meet the requirements of modern information systems. The operation of these devices is based on the raster method of element-by-element synthesis of a light line, combined with the photoelectric principle of image** recording on paper [1]. Now, the most widely used are printers in which images of symbols on the surface of a photosensitive layer of an electrographic cylinder are formed by a semiconductor laser using a mechanical (mirror or prism) scanning system [2]. Concurrently, in recent years devices were developed whose characteristic feature is that an entire line of symbols is formed using multichannel light emitting [3] or light modulating [4] devices without mechanical or any other type of scanning components. A promising direction in the development of such devices involves the use of electrooptic ceramics-based one-dimensional time-space light modulators (TSLMs) for symbol formation [5]. Among the advantages of tyis type of TSLM are first of all high light modulation characteristics (contrast of at least 100), high speed (1-100 kHz clock rate of line-by-line image formation) and a wide spectral range of light modulation (0.4-0.8 μm), which makes it possible to match spectral characteristics of a TSLM optical image to the sensitivity region of series-produced electrographic drums [6 and 7].

The format of one-dimensional TSLMs used for symbol synthesis in an optical printer must match the width of a printed page. As a result, the total number of light valves at a 10 points per mm resolution must be equal to 2×10^3. In this case, it is impossible to use parallel methods for time-space light modulation [6 and 7], which consist in simultaneous application of control potentials to all light valves and require the corresponding number of electronic control channels.

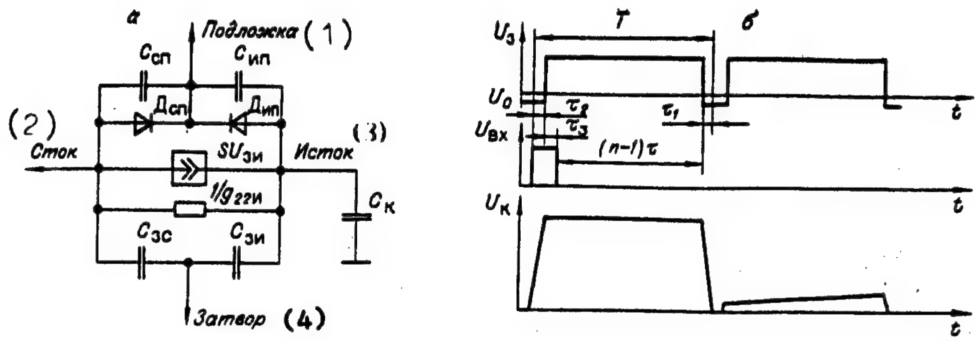


Figure 1

Key:

1. Substrate
2. Drain
3. Source
4. Valve

In this respect, of interest is the development of hybrid TSLM with multiplex control, which make it possible to organize a "dynamic memory" mode in an electrooptic material, use series-parallel methods for time-space light modulation and reduce the number of electronic control channels. To organize the "dynamic memory" mode, addressing circuits of a multiplex TSLM must include a switching element - a MOSFET. Using a MOSFET, charges are injected into the valve area and prolonged relaxation of valves space charges during the addressing cycle of the entire TSLM is achieved. The objective of this work is to study multiplex methods for time-space light modulation in hybrid TSLMs based on electrooptic ceramics-MOSFET structures.

Analysis of Parameters of Electrooptic Ceramic -MOSFET Structures. An equivalent circuit of the structure is shown in Figure 1,a [8]. Light valves of a multiplex hybrid TSLM are divided into n groups of m valves each. Same-name drains of MOSFETs in each group are connected to corresponding information inputs, and their valves are connected to corresponding address inputs of the TSLM. Information is entered into the TSLM when an m-register information word is sent to drains of MOSFETs of the addressed group and resolution potential $U_3 = -U_0$ (U_0 is threshold voltage) is applied to their shutters. A time diagram of the operation of the electrooptic ceramics-MOSFET structure is shown in Figure 1,б. Addressing time t of a valve group includes time t_1 of erasing information entered during the preceding TSLM addressing cycle, information recording time t_2 and delay t_3 determined by the setting of transient processes ($t = t_1 + t_2 + t_3$). The total TSLM addressing period determined by number n of group shutters is T = nt.

In the process of addressing the structure ($U_3 = -U_0$, $U_C = U_{BX}$), valve self-capacitance C_K is charged. Voltage U_K across the shutter is derived from the following formula:

$$U_к = U_{вх}\{1 - \exp[-t_2/(C_к + C_{ип} + C_{зи})R_0]\}. \qquad (1)$$

To store information in a light valve, it is necessary to maintain the level of the input signal $U_{вх}(0 \leqslant U_{вх} \leqslant U_{вх}^{max})$ at a MOSFET drain during delay time t_3, which depends on the duration of processes of restoration of resistance of the closed channel "drain-source". When switching to the information storage mode, voltage across the shutter changes by $\Delta U_з = U_{вх}^{max} + U_0$. This is accompanied by processes of charge transfer through capacitance $C_{зи}$ and milking of capacitances C_K and $C_{ип}$. Because of this, voltage across the valve will be equal to

$$U'_к = U_к + \Delta U'_к, \text{ где } \Delta U'_к = \frac{C_{зи}}{C_к + C_{ип}} (U_{вх}^{max} + U_0). \qquad (2)$$

We shall now examine the mode of storage of recorded information in an electrooptic ceramic -MOSFET structure. The stability of storing light transmission levels during period (n - 1)t of addressing the remaining valve groups can change due to the following factors:

milking of valve self-capacitance C_K through the closed channel source-drain $(U_з = U_{вх}^{max})$ (the worst-case scenario is when a logical "0" ($U_C = 0$) is recorded in a light valve of the first group, while a logical "1" is applied to the MOSFET drain during (n - 1) clock cycles of addressing the remaining groups $(U_c = U_{вх}^{max})$

discharge of capacitance C_K through the closed channel "source-drain" (the worst-case scenario is when a logical "1" ($U_C = U_{вх}^{max}$)) is recorded in a valve of the first group, while a logical "0" is applied to the MOSFET drain during (n - 1) clock cycles of the remaining groups ($U_C = 0$);

leakage currents over the MOSFET surface and through valve resistance; and

milking of capacitance C_K through the reverse-biased diode "substrate-source" $(U_з = U_п = U_{вх}^{max})$.

The first three mechanisms have no significant effect on light modulation characteristics, because for a typical MOSFET with an induced channel resistance of a completely closed channel "drain-source" is over 10^{10} Ohm, resistance between MOSFET terminals is over 10^{12} Ohm [8], and valve resistance (electrooptic ceramics monocrystal) is over 10^{14} Ohm.

TSLM contrast characteristics are significantly affected by milking of capacitance C_K through the reverse-biased diode "substrate-source". Voltage change across valve capacitance C_K during storage time (n - 1)t is derived from the following expression:

$$\Delta U''_к = U_{вх}^{max}\{1 - \exp[-(n-1)t/(C_к + C_{ИП} + C_{ЗИ}) R_{обр}]\}. \qquad (3)$$

The value of $R_{обр}$ depends in turn on voltage $U_{ИП}$. In the storage of a logical "0" mode the value of $(U_к = 0, U_п = U_{вх}^{max})$ $R_{обр}$ is minimal. For other values of $0 < U_к < U_{вх}^{max}$ $R_{обр}$ increases, and the corresponding change of valve light transmission, which depends on parasitic milking of capacitance C_K, decreases. Thus, levels with a lower value of light transmission turn out to be less stable.

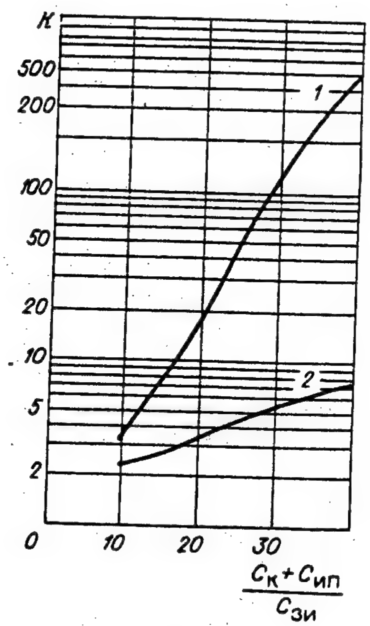


Figure 2

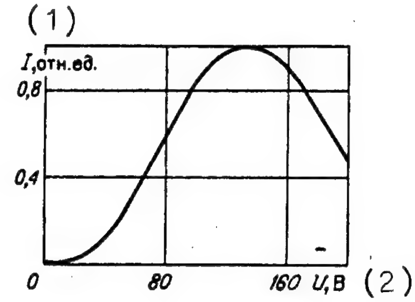


Figure 3

Key:
1. Relative units
2. V

As a result, total voltage change across a light valve, derived from (2) and (3), is

$$\Delta U_{к} = \Delta U'_{k} + \Delta U''_{к}. \qquad (4)$$

A change in recorded signal amplitude U_K leads to decreased contrast of an optical image. Let an electrooptic material have light modulation characteristic I(U), where I is light flux intensity. Based on (2), (3) and (4), optical image contrast K can be presented as the ratio of energies of light fluxes that have passed through the valve during the information storing time in TSLM, in cases of recording logical "1" and "0":

$$K = E^{«1»}/E^{«0»} = \int_0^{(n-1)t} I\left(U_{вх}^{max}\right) dt \Bigg/ \int_0^{(n-1)t} I\left(\Delta U_{к}\right) dt = I\left(U_{вх}^{max}\right) \Bigg/ \left[I'\left(\Delta U'_{к}\right) + \frac{1}{(n-1)t} \int_0^{(n-1)t} I''\left(\Delta U'_{к}, \Delta U''_{к}\right) dt \right], \qquad (5)$$

where $I'(\Delta U'_{к})$ is parasitic light transmission of the valve due to charge transfer through capacitance $C_{зи}$ from expression (2), and $I''(\Delta U'_{к}, \Delta U''_{к})$ is parasitic light transmission of the valve that depends on milking of capacitance C_K via a reverse-biased diode "substrate-source" from expression (3).

Figure 2 shows dependences of contrast K of electrooptic material-MOSFET structures. Curve 1 was plotted for structures based on electrooptic ceramic TsTSL, composition 9/65/35, and curve 2 was plotted for $LiNbO_3$ monocrystals [9]. The quadratic electrooptical effect (Figure 3) was used for light modulation in ceramic TsTSL, and the linear electrooptic effect was used for light

modulation in $LiNbO_3$. Curves in Figure 2 were plotted based on expression (5), taking into account MOSFET parameters presented below. The information change period in TSLM was chosen equal to 1 ms. In both cases, signal amplitude $U_{вх}^{max}$ was 100 V.

One can see from the analysis of results in Figure 2 that values of contrast K depend on the type of light modulation characteristic of the electrooptic material. In the case of the linear electrooptical effect (curve 2), processes of charge transfer through capacitance $C_{зи}$ and milking of valve self-capacitance via the reverse-biased diode "substrate-source" lead to the generation of a parasitic charge on the light valve and substantial contrast drop. Even increasing valve self-capacitance C_K to 100 pF and up (a typical value of MOSFET $C_{зи}$ is approximately 1 pF) does not make it possible to obtain contrasts over 10-20.

It is interesting to examine the possibility of using multiplex hybrid TSLMs of this type made of one of the most widely used light modulating materials - liquid crystals. They have a clearly pronounced threshold characteristic [9], however, self-capacitance of a liquid-crystal valve is equal to 3-5 pF. Based on (2), (3) and (5), the corresponding value of contrast K in this case does not exceed 2. To increase contrast, additional reservoir capacitors [10] must be used; however, this makes the device much more complicated.

Electrooptic ceramics is the most promising material for use in hybrid TSLMs with multiplex addressing. High contrast values (see curve 1 in Figure 2) are due to an advantageous combination of threshold characteristics of the quadratic electrooptical effect and high value of valve self-capacitance (approximately 100 pF).

Experimental Results. Experimental studies of electrooptic ceramic -MOSFET structures were conducted based on a 128-channel TSLM made of TsTSL ceramics composition 9/65/35. For light modulation the lateral effect of induced dual-beam light refraction was used. Designwise, the TSLM incluces a TsTSL ceramic plate, approximately 0.2 mm thick. A system of metal electrodes was formed on the plate surface, using photolithography methods. Metal electrodes were 35 μm wide and spaced at 75 μm. The plate was placed between crossed polarized films and glued to a glass substrate with a transparent silicon-organic glue. The TSLM light modulation characteristic at a 0.44 μm wavelength and 1 kHz control pulse frequency is shown in Figure 3. Contrast of the TSLM optical image was at least 100. Valve self-capacitance C_K varied between 124 and 109 pF in the 0-100 V voltage range.

Silicon plates KEF-7.5 were used as semiconductor material for MOSFETs with a p-type induced channel. The thickness of the subshutter dielectric was approximately equal to 0.4 μm. MOSFET output characteristics are shown in Figure 4. It had the following electrical parameters: threshold voltage approximately 25 V, initial drain current 0.3 mcA at $U_{си}$ = 100 V, channel resistance R_0 10 kOhm, breakdown voltages across the terminals approximately 120 V, capacitance $C_{зи} \sim$ 1 pF, and capacitance $C_{пп}$ varied between 6 and 2.5 pF in voltage range $U_{нп}$ = 0-100 V.

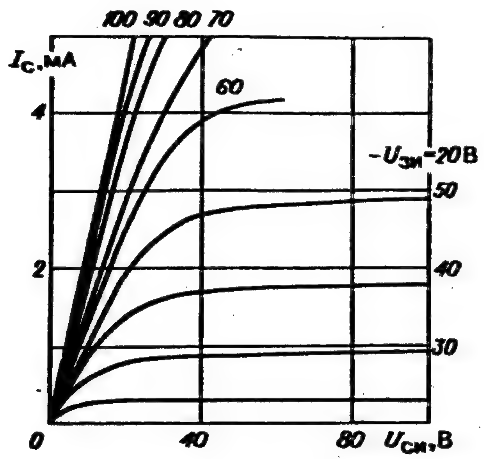


Figure 4

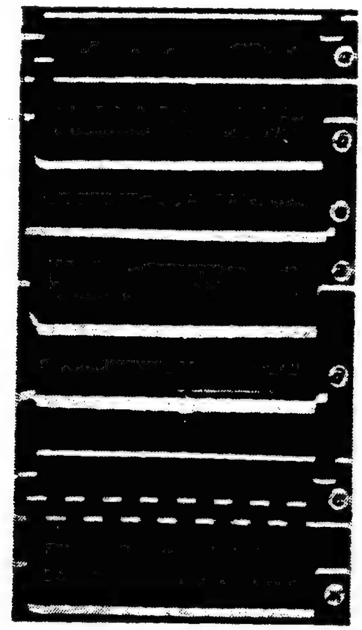


Figure 5

Oscillograms in Figure 5 illustrate the operation of an electrooptic ceramic-MOSFET structure. Vertical scales are as follows: a and e - 200 V per graduation; б, г, ж - 100 mV per graduation; and b and д - 5 mV per graduation. Horizontal scales a- ж are 0.5 ms per graduation. Figure 5,a shows shutter and drain voltage waveforms. Figure 5, б characterizes the efficiency of storing a recorded logical "1" $(U_к=U_п=U_{вх}^{max})$. The stability of the level of storing a logical "1" is due to high resistance of the closed channel "drain-source" in the discharge circuit of capacitance C_K and to milking of C_K through the reverse-biased diode "source-substrate" (see Figure 1,a). The effect of parasitic milking of capacitance C_K on optical image contrast during the time of storage of a logical "0" is shown in Figure 5,b. Charge transfer through capacitance $C_{зи}$ according to (2) is practically not observed, because in this case $(C_к+C_{кп})/C_{зи}>100$ and parameter $\Delta U_к'$ from (2) is below the threshold value on the light modulation characteristic of TsTSL ceramics (see Figure 3). The observed increase in partial light transmission of the valve during the information storage time is only due to milking of capacitance C_K via the reverse-biased diode "source-substrate" in accordance with (3) and (5).

The effect of charge transfer through capacitance $C_{зи}$ on TSLM light modulation characteristics is shown in oscillograms in Figure 5, г, д. In this case, ratio $(C_к+C_{кп})/C_{зи}$ is reduced artificially to 4, which results in a noticeable change in light transmission when recording signals with amplitudes $0{,}8U_{вх}^{max}$ (Figure 5, г) and U_{BX} = 0 (Figure 5, д).

Oscillograms in Figure 5,e (voltage waveforms across the shutter and drain) and 5, ж $(U_c=U_{вх}^{max})$ illustrate the stability of maintaining the light transmission level when addressing other valve groups of a multiplex TSLM.

Conclusion. This work presents a study of hybrid TSLMs with multiplex control based on electrooptic material-MOSFET structures, which make it possible to organize the mode of "dynamic memory" of light transmission levels of light valves during the addressing cycle. Mechanisms of charge transfer in the process of addressing the structures and the effect of these mechanisms on TSLM contrast characteristics have been examined. The possibility of using various electrooptic materials ($LiNbO_3$ monocrystals, electrooptic ceramics and liquid crystals) has been evaluated. Based on theoretical and experimental studies, it has been established that the use in electrooptic material-MOSFET structures of transparent ceramics of the TsTSL system, which has the threshold characteristic of the quadratic electrooptical effect and a high value of valve self-capacitance, makes it possible to obtain optical image contrast of at least 100. The obtained results can be used in developing one-dimensional TSLMs with a large number (10^3-10^4) of light modulation channels.

BIBLIOGRAPHY

1. Chepenko, V.G., Dangovene, G.K. and Batalov, I.N., "Development of Electrophotographic Laser Printers", POLIGRAFICHESKAYA PROMYSHLENNOST: OBZORNAYA INFORMATSIYA, 1986, No 8.

2. Vul, V.A., "Laser Devices for Readout of Computer Graphics and Alphanumeric Information", ZARUBEZHNAYA ELEKTRONIKA, 1978, No 5.

3. "Light-Emitting Diodes - the Laser's Competitor in Printers", ELEKTRONIKA, 1984, Vol 57, No 7.

4. "Printer With Liquid Crystal Shutters", Ibid., No 25.

5. Pilipovich, V.A., Guk, A.V., Kolennikov, P.I. et al., "One-Dimensional TsTSL Ceramic TSLMs in High-Speed Optical Printers", in book "Polucheniye, issledovaniye i primeneniye prozrachnoy segnetokeramiki: Tezisy dokladov 3-20 Mezhduvedomstvennogo seminara (Riga, 1988)" [Manufacturing, Study and Application of Transparent Ferroelectric Ceramics: Theses of Papers of 3-20 Interbranch Seminar (Riga, 1988)], Riga, 1988.

6. Pilipovich, V.A., Guk, A.V., Kolennikov, P.I. et al., "TsTSL Ceramics-Based Linear Controlled Transparencies", AVTOMETRIYA, 1981, No 5.

7. Pilipovich, V.A., Guk, A.V., Malakhovskiy, V.R. et al., "TsTSL 9/65/35 Transparent Ceramics-Based Multichannel Electrooptical Light Modulator", PTE, 1987, No 2.

8. "Polevyye tranzistory. Fizika, tekhnologiya i primeneniye" [Field Transistors: Physics, Technology and Application", S.A. Mayorov editor, translated from English, Moscow, "Sovetskoye radio", 1971.

9. Vasilyev, A.A., Kasasent, D., Kompanets, I.N. and Parfyonov, A.V., "Prostranstvennyye modulyatory sveta" [Spatial Light Modulators], Moscow, "Radio i svyaz", 1987.

10. Pilipovich, V.A., Guk, A.V., Kolennikov, P.I. et al., "Device for Information Input Into Holographic Memory Based on Controlled Liquid Crystal Mosaic Transparency", AVTOMETRIYA, 1979, No 1.

UDC 535.241.13:681.527

Multichannel TsTSL-Ceramic TSLMs in Devices for Input of Multigradation Images Into Computers

907G0094D Novosibirsk AVTOMETRIYA in Russian No 6, Nov-Dec 1989 (manuscript received 7 Apr 88) pp 75-80

[Article by A.V. Guk under the "Physical Aspects of Micro- and Optoelectronics" rubric]

[Text] The efficiency of application of digital image processing systems depends to a large degree on successful solution of problems of input of multigradation images into computers. The input operation includes conversion of an analog image into an equivalent digital array (image digitizing), which is then processed in the computer in accordance with a given program. Image conversion into a digital array is done as a result of measurement of certain physical characteristics, e.g. brightness and optical density.

At present, hardware development for high-quality high-speed input of multigradation images is an urgent problem. According to some authors' estimates [1 and 2], such devices should provide input into a computer of 10^5 discrete points per second at over 100 quantization levels of brightness or optical density. The speed of available single-channel scanning photodetectors [3] is insufficient. ERT-based systems have limited application, due to instability of the position, dimensions and brightness of a light spot on the screen, which results in several dozen quantization levels [2]. Input devices based on multielement light emitters and photodetectors, for instance, strips of light emitting diodes and devices with charge coupling, are characterized by considerable variation of parameters of individual elements, which reduces the number of distinguishible gradations in an analyzed image.

This work studies problems of using multichannel TSLMs [time-space light modulators] of the "spread matrix" type based on a system TsTSL quasiferroelectric ceramics, composition 10/65/35, for input of images into computers. In this device, a TSLM operates in the scanning mode with clock frequency of 0.1-1 MHz. Images of all TSLM light valves (LV) are projected onto a single photodetector [4], which considerably simplifies electronic control circuits of the readout channel. The use of a multichannel TSLM makes it possible, with the same frame format and degree of discretization, to reduce substantially the number of necessary points for positioning the

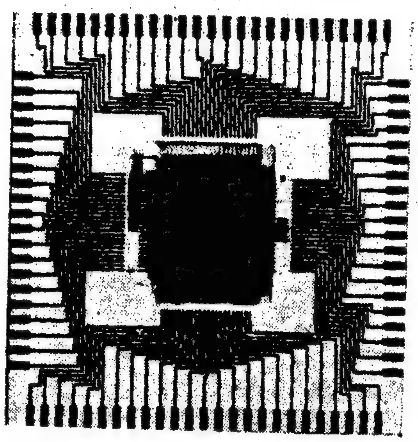


Figure 1

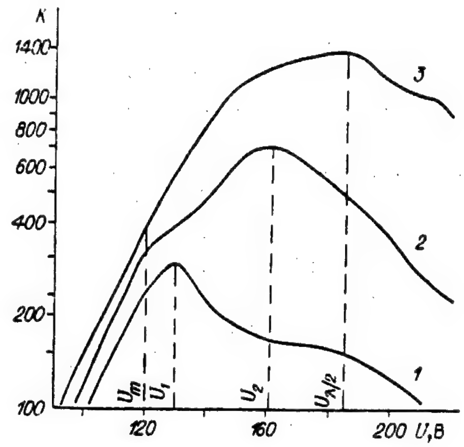


Figure 2

Key:
1. V

scanning mechanism, compared to a single-channel photodetector, and increase the output of the entire device.

Figure 1 shows the exterior of a matrix TSLM of the "spread matrix" type with 44 x 44 LVs. A TSLM sample consists of two 0.2 x 10 x 15 mm TsTSL ceramic plates with systems of comb-type electrodes. The electrodes are 100 μm wide and located 200 μm apart. The TsTSL ceramic plates were placed orthogonally to each other so that electrode systems on both plates formed rows and columns. The plates were placed between three crossed polarized films and glued to a glass substrate with a transparent silicon-organic glue.

For light modulation one used the lateral electrooptical effect of induced two-beam refraction. Designwise, a matrix TSLM consists of two mutually orthogonal linear TSLMs with a slot LV aperture. An LV is turned on when potential difference is applied to electrodes confining its aperture. Figure 2 shows dependences of contrast characteristics of a linear TSLM in the element-by-element scanning mode on voltage applied to a turned-on LV, for light wavelength λ = 440 nm and clock frequency 10 kHz. Light modulation characteristics were studied using a methodology of dynamic studies of multichannel TSLMs that makes it possible to suppress the effect of accumulation of residual polarization. The TSLM was controlled by means of bipolar voltage pulses in the shape of symmetric meanders. Curve 1 in Figure 2 is the dependence of contrast K_1 of valves located next to the turned-on valve, and curve 2 is the dependence of contrast K_2 of the next pair of adjacent LVs. Curve 3 describes the dependence of contrast K of the remaining valves of a linear TSLM. One can see from this graph that first, as voltage U increases, K_1, K_2 and K increase monotonically, due to increasing transmission of the turned-on LV in accordance with the modulation characteristic of the quadratic electrooptical effect in TsTSL ceramics. Then, despite further increase in transmission, the rate of increase of contrast K_1 slows down, and at certain voltage $U_1 < U_{\lambda/2}$ K_1 reaches its maximum. This behavior of curve

$K_1(U)$ is due to partial switching of an adjacent turned-off LV. At low voltages, the degree of the switching is low and does not affect an increase in K_1 when transmission of the turned-on LV increases. At high voltages, partial transmission of the adjacent valve starts increasing faster, thus slowing down the increase in K_1. At $U = U_1$, the parasitic electric field in this LV's aperture becomes so high that its partial transmission starts increasing faster than that of the turned-on LV. In the process, contrast K_1 decreases.

The behavior of curve $K_2(U)$ is similar to that of $K_1(U)$, except that the maximum of K_2 is achieved at $U_2 > U_1$, because the second adjacent valve is located further from the turned-on one, and the induced parasitic electric field in its aperture is more strongly attenuated. For other LVs, regardless of their position relative to the turned-on LV, contrast over the entire voltage range is described by one curve - curve 3. In this case, a change in contrast K is only due to a change in light transmission of the turned-on LV and the type of light modulation characteristic of the electrooptic material.

We shall now examine the mode of input of multigradation images into computers, using a matrix TSLM with line scanning and a single photodetector. A matrix TSLM with n^2 LVs consists of two orthogonal linear TSLMs with a slot LV aperture that have contrast characteristics per Figure 2. We shall assume that at a certain moment in time $t_{i,j}$ a number j slot LV of the first linear TSLM and the i-th LV of the second linear TSLM are turned on. Then, one can present transmittance of the first columnar linear TSLM as follows:

$$C^1_{i,j} = 1, \qquad j = 1, \ldots, n;$$
$$C^1_{i\pm m,j} = \begin{cases} 1/K_1, & m = 1; \\ 1/K_2, & m = 2; \\ 1/K, & m \geqslant 3. \end{cases} \tag{1}$$

Similarly, transmittances of LVs of the second row linear TSLM are derived as follows:

$$C^2_{i,j} = 1, \qquad i = 1, \ldots, n;$$
$$C^2_{i,j\pm m} = \begin{cases} 1/K_1, & m = 1; \\ 1/K_2, & m = 2; \\ 1/K, & m \geqslant 3. \end{cases} \tag{2}$$

We shall assume that the dynamic range of intensities of light fluxes passing through various sections of a photograph is within $[I_{min}, I_{max}]$ and that light flux with intensity I_a ($I_{min} < I_a < I_{max}$) is incident on the turned-on LV of the matrix TSLM. Then, one can present light intensity at the output of the turned-on LV as $I_{i,j} = \eta I_a$, where η is optical efficiency of the TSLM. The signal detected in this case by a photodetector will consist of the sum of light fluxes that have passed through the i-th, j-th LV and all other valves in accordance with their contrast properties. An additional light flux due to finite values of matrix TSLM contrasts is parasitic in character, and it introduces an error in the determination of the intensity of the sought light

flux passing through the turned-on LV. Here, two limiting cases are possible; they correspond to light incidence with equal intensity, I_{min} in the first and I_{max} in the second case, onto all LVs except the i-th, j-th one. In these cases, one can present signals detected by a photodetector as follows:

$$I^{\Phi\Pi}_{a\,min} = \eta\left(I_a + I_{min}\sum_{1}^{n^2-1} C_{i,j}\right);$$
$$I^{\Phi\Pi}_{a\,max} = \eta\left(I_a + I_{max}\sum_{1}^{n^2-1} C_{i,j}\right). \quad (3)$$

Transmittances of a matrix TSLM can be presented in the form of the product of corresponding transmittances of the first (1) and second (2) linear TSLMs:

$$C_{i,j} = C^1_{i,j}C^2_{i,j}. \quad (4)$$

For the matrix TSLM under consideration (see Figure 2), experimental values of contrasts are $K > K_2 > K_1 > 100$. Then, ignoring second-order infinitesimal (compared to $1/K_1$, $1/K_2$ and $1/K$) terms, one can present expression (4) as follows:

$$C_{i,j} = \frac{4(K_1+K_2)(K+n)}{K_1K_2K} + \frac{2(n-5)}{K} + \frac{n(n-10)}{K^2}. \quad (5)$$

Let light flux intensity at the input of the i-th, j-th LV change by $\Delta I > 0$ and become equal to $I_b = I_a + \Delta I$, wherein $I_{min} < I_b < I_{max}$. Then, the respective maximum and minimum signals arriving at a photodetector can be presented in the following form:

$$I^{\Phi\Pi}_{b\,min} = \eta\left(I_a + \Delta I + I_{min}\sum_{1}^{n^2-1} C_{i,j}\right);$$
$$I^{\Phi\Pi}_{b\,max} = \eta\left(I_a + \Delta I + I_{max}\sum_{1}^{n^2-1} C_{i,j}\right). \quad (6)$$

Because the input signal has increased, i.e. $I_b > I_a$, it is obvious that signal $I^{\Phi\Pi}_b$ at the photodetector will always exceed $I^{\Phi\Pi}_a$. In particular, the following condition must be satisfied:

$$I^{\Phi\Pi}_{b\,min} \geqslant I^{\Phi\Pi}_{a\,max}. \quad (7)$$

Substituting appropriate expressions from (3) and (6) into inequality (7), we derive after identical transformations that

$$I_{max} - I_{min}/\Delta I \leqslant \left(\sum_{1}^{n^2-1} C_{i,j}\right)^{-1}. \quad (8)$$

The left-hand side in expression (8) is actually the number of quantization levels m of an image with dynamic range I_{max}, I_{min} and quantization step ΔI. Equality in expression (8) characterizes the limiting case, when boundaries of two adjacent gradations of the gray shade scale converge and the maximum

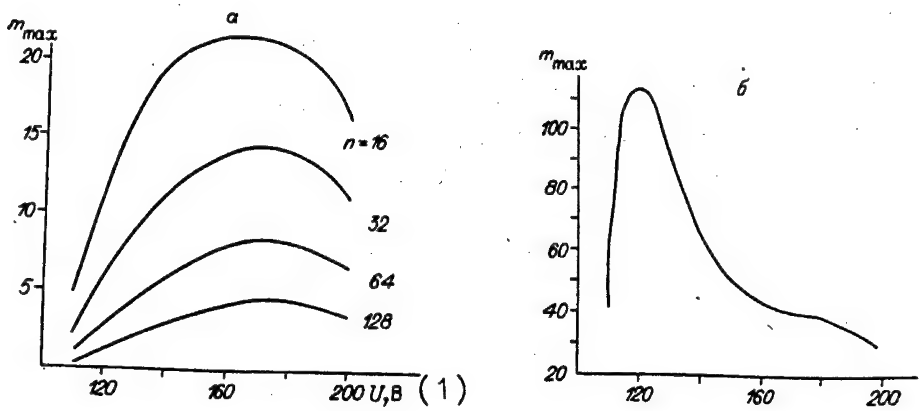


Figure 3

Key:
1. V

possible number of quantization levels m is reached. Taking into account (5), we can write

$$m_{max} = \left[\frac{4(K_1+K_2)(K+n)}{K_1K_2K} + \frac{2(n-5)}{K} + \frac{n(n-10)}{K^2}\right]^{-1}. \qquad (9)$$

Figure 3,a shows families of curves that express the dependence of m_{max} on U at various n. The curves were plotted based on expression (9) and contrast characteristics of linear TSLMs (see Figure 2). Obviously, if it is necessary to obtain a certain number of intensity gradations in the analyzed image, the possible number of LVs in a matrix TSLM must not exceed a certain value that decreases as m_{max} increases. Thus, for λ = 440 nm at m_{max} = 4 a matrix TSLM can have no more than 128 x 128 LVs. At m_{max} = 14 TSLM information capacity does not exceed 32 x 32. And the character of m_{max} dependence on n is such that even when the number of LVs is low, e.g. 16, the maximum possible number of quantization levels of an analyzed image does not exceed 21 and does not satisfy conditions of a large number of image processing problems, when the required value of m_{max} is 64 or higher [3].

We shall now analyze expression (9). The main restriction on the number of gradations of the gray shade scale is imposed by terms 2n/K and n^2/K^2. Term 2n/K defines parasitic light flux passing through valves of the i-th row and j-th column of a matrix TSLM, and term n^2/K^2 is light flux through LVs numbers $i \pm m$ and $j \pm m$, where $m \gg 3$. One can increase the number of quantization levels of the input image by using a differential input mode [4]. In our case, the input mode is modified as follows. First, total signal intensity at the photodetector when the i-th, j-th LV is turned on is measured. Then, intensities of two light fluxes passing through the matrix TSLM are subtracted from this signal when only the i-th row of the first linear TSLM is turned on (all columns of the second linear TSLM are turned off) and the j-th column of the second linear TSLM is turned on (the lines of the first linear TSLM are turned off). Then, one adds to this the light flux that has passed through the turned-off matrix TSLM. By

analogy with (1), (2) and (4), matrix TSLM transmittances in those modes where only one linear TSLM is turned on can be presented as follows:

$$C_{i,j}=1/K,\quad j=1,\ldots,n;\quad C_{i,j}=1/K, i=1,\ldots,n;$$
$$C_{i\pm m,j}=\begin{cases}1/K_1K, m=1;\\ 1/K_2K, m=2;\\ 1/K^2,\quad m\geqslant 3;\end{cases}\quad C_{i,j\pm m}=\begin{cases}1/K_1K, m=1;\\ 1/K_2K, m=2;\\ 1/K^2,\quad m\geqslant 3.\end{cases}\qquad (10)$$

Total transmittance of a matrix TSLM in both cases is

$$\sum_1^{n^2} C_{i,j}=\frac{n}{K}+\frac{2n}{K_1K}+\frac{2n}{K_2K}+\frac{n(n-5)}{K^2}.\qquad (11)$$

Transmittance of a turned-off TSLM is

$$\sum_1^{n^2} C_{i,j}=\frac{n^2}{K^2}.\qquad (12)$$

In the differential image input mode under consideration, the intensity of the light flux passing through the analyzed i-th, j-th LV will change; based on (3), it can be presented in the following form:

$$I_{i,j}=\eta I_a\left(1-\frac{2}{K}+\frac{1}{K^2}\right).\qquad (13)$$

For this signal, we shall conduct analysis similar to the above presented one (expressions (3)-(8)); based on (10)-(13) and ignoring second order infinitesimal (compared to $1/K_1$, $1/K_2$ and $1/K$) terms, we derive:

$$m_{\max}=\frac{K_1K_2(K-2)}{4(K_1K+K_2K-2K_1K_2)}.\qquad (14)$$

Thus, in a differential input mode the number of image quantization levels only depends on contrast properties of a matrix TSLM and does not depend on the number of LVs in the latter.

Figure 3, б shows dependences of m_{max} on control voltage U; they are plotted based on formula (14) and contrast characteristics of linear TSLMs (see Figure 2). Comparing curves in Figures 3,a and б, one can see that in a differential input mode the possible number of quantization levels increases considerably. While in a direct input mode for λ = 440 nm m_{max} did not exceed 21 when the number of LVs was as low as 16 x 16, the number of quantization levels in a differential mode for any number of LVs exceeds 100.

Of interest is the shape of the $m_{max}(U)$ curve. It has a well-defined maximum at operating voltage $U_m=(0{,}6-0{,}7)\,U_{\lambda/2}$. A simple analysis of expression (14) demonstrates that m_{max} depends both on absolute values of K_1, K_2 and K

and on the variation of their values. An increase in absolute values leads to an increase in m_{max}, while increased variation leads to its reduction. One can see from data in Figure 2 that m_{max} is reached at control voltage $U = U_m$, which is characterized by small variation of values of K_1, K_2 and K. Further increase in absolute values of contrasts can no longer compensate for an increase in their variation, and the value of m_{max} decreases when control voltages increase.

It should be noted that at $U = U_m$ transmittance of a turned-on LV is approximately 30% of its maximum value at $U = U_{\lambda/2}$. At $U > U_m$, transmittance increases, but the possible number of input image gradations drops from 114 down to 38 (for $U = U_{\lambda/2}$). i.e., when operating at half-wave voltage, a matrix TSLM has good light modulation characteristics, but the number of quantization levels in this case is approximately one-third of the maximum possible number.

Thus, based on experimental studies of contrast properties of matrix TSLMs made of a system TsTSL electrooptic ceramics, modes of image input into computers have been recommended. The obtained results suggest the possibility of efficient application of such devices for input of multi-gradation images into computers.

BIBLIOGRAPHY

1. Grishin, M.P., Kurbanov, Sh.M. and Markelov, V.P., "Avtomaticheskiy vvod i obrabotka fotograficheskikh izobrazheniy na EVM" [Automatic Input and Computerized Processing of Photographic Images], Moscow, "Energiya", 1976.

2. Vaskov, S.T., Mamontov, G.M, Potashnikov, A.K. and Tkach, S.Ye., "Skaniruyushchiye ustroystva na ELT vysokogo razresheniya" [Scanning Devices With High-Resolution ERTs], Novosibirsk, "Nauka", 1978.

3. Pratt, W., "Tsifrovaya obrabotka izobrazheniy" [Digital Image Processing], Moscow, "Mir", 1982.

4. Pilipovich, V.A, Cuk, A.V., Malekhovskiy, V.P. et al., "Elektroopticheskoye ustroystvo vvoda/vyvoda izobrazheniy" [Electrooptic Device for Input/Output of Images], USSR Certificate of Authorship No 1323998, published in BI [Bulletin of Discoveries, Inventions and Trade Marks], 1987, No 26.

UDC 621.325.5-181.4

Design Principles for Control-Friendly Microprocessor Systems

907G0110A Moscow AVTOMATIKA, TELEMEKHANIKA I SVYAZ in Russian No 2, Feb 90, pp 21-24

[Article by V. V. Sapozhnikov and Vl. V. Sapozhnikov]

[Text] On the Quality of Self-Checking

A system is known as control-friendly if measures are provided in its design to simplify the process of detection and correction of faults in its internal elements. In practice, all systems possess this quality more or less. For example, such measures in an electrical centralization system include: block design of the system, indication on the station supervisor panel, possibility of visual checking of the condition of relays, and so on. In a certain sense, relay systems have "natural" control-friendliness, since it is easy to check the working of a relay. However, systems built from microelectronic components do not have this quality and it must be provided by special methods.

The prime method is design of systems with automatic fault detection. This makes it possible either to disconnect the system when the very first fault appears or to switch the control to a standby layout. The advantage of systems with fault detection is that they have less redundancy as compared to fault-resistant systems. Their use is most expedient in the design of attended systems with lengthy service terms (such as are almost all railway automation and telemechanics systems). The synthesis of such systems in a block layout with signalling of malfunction for each block enables quick correction of damage while the system is operating by replacement of a malfunctioning block.

In the present article we examine one of the most promising trends in the handling of this problem--the design of discrete systems with self-checking, or self-checking systems. The quality of self-checking consists in the following. In order to detect faults, the discrete system is provided with a checking circuit, the purpose of which is to form a special error signal when a malfunction occurs. This signal is used to disconnect the faulty system from the object being controlled and to turn on the standby layout. Redundancy, which is the price paid for the quality of fault detection, is introduced into the internal structure of the discrete system. As a result, the control-friendliness of the system is enhanced and it acquires such properties as allow easy detection of the occurrence of malfunctions by means of the checking circuit.

Let us define the properties of this type of system. To illustrate these properties, we consider the following example. Figure 1 shows a relay-contact circuit designed to ascertain the passage of the axle of rolling stock through a check point, equipped with a contact transmitter Д. The output v generates a signal each time that an even-counted axle passes across the check point. We remark that the relay-contact circuit is considered here only for reasons of simple and graphic presentation of the ideas of control-friendliness. Everything to be said hereafter also applies to the logic apparatus built from integrated technology.

Figure 1 also shows the operating diagram of the circuit, having four internal states: Y_1Y_2 = 00, 01, 11, 10. The signal at output v is generated when the circuit arrives in condition 10. Obviously, in event of malfunctioning of relays Y_1 and Y_2, the specified operating algorithm of the circuit may be distorted (a false signal may appear at output v). It is also evident that this circuit does not possess the quality of control-friendliness, since its functioning uses all four possible internal states and therefore it cannot be determined at each instant whether the circuit is in the given state as a result of proper functioning or as a result of malfunctioning of a particular element.

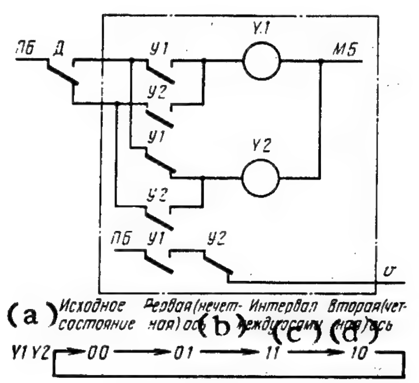


Fig. 1

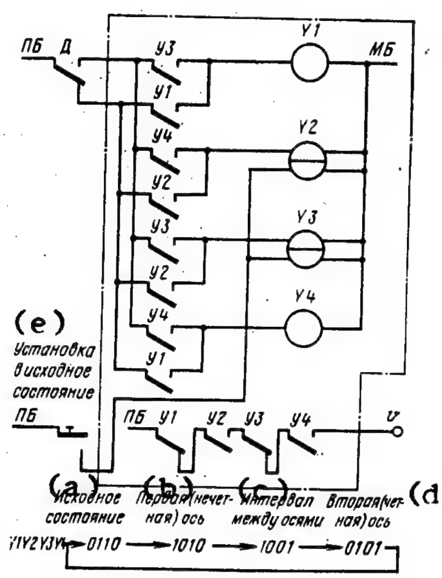


Fig. 2

Key:

a. Initial state
b. First (odd) axle
c. Interval between axles
d. Second (even) axle
e. Setting in initial state

In order to impart to the circuit the quality of control-friendliness, it is necessary to introduce redundant internal states of the system. This requires increasing the number of relays. It then becomes possible to discern, among the total set of internal states of the system, the set of basic (working)

states S and the set of error (protective) states S_R, recognizing that the properly working system will always be in one of the basic states, and any malfunction of the given class will switch the system to one of the error states. The fact of switching the circuit from the set of basic states S to the set of error states S_R may be regarded as an indicator of faultiness. The transition itself may be detected by means of a special checking circuit.

Figure 2 shows a layout that handles the same task as the layout in Fig. 1, but possesses the quality of control-friendliness. The circuit is built from four relays and, consequently, has 16 internal states. Four of these are the basic states. These are shown in Fig. 2 in the transition diagram. The signal at the output v is generated when the circuit arrives in state 0101. All other states belong to the set of error states S_R. Any single malfunction of the relays Y_1-Y_4 will switch the circuit from the basic state at the given moment to one of the error states. For example, if the circuit is in initial state 0110 (when the rear contact of the transmitter П is closed) and a malfunction of type $1 \rightarrow 0$ occurs in relay Y_2 (the relay should be activated, but because of the fault it is disconnected), then the circuit will first switch to state 0010 and immediately thereafter to state 0000 (since there is an interruption in the activation circuit of relay Y_3, which travels across the rear contact of the transmitter П and the front contact of relay Y_2). In this state the system is disabled, since the activation circuits of all relays Y_1-Y_4 only pass across the front contacts.

If a malfunction of type $0 \rightarrow 1$ of relay Y_1 occurs when the system is in initial state 0110 (the relay should be disconnected, but because of the fault it is activated), the system will first switch to state 1110, and immediately thereafter to state 1111 (since a circuit is formed to activate relay Y_4, passing across the rear contact of the transmitter П and the front contact of relay Y_1). In this state, the system is also disabled, since all contacts of the internal relays making up the circuit are closed.

A feature of this circuit is the fact that all its basic states have a single distinguishing characteristic. In each of these states, two of the four relays are activated and two are disconnected. As has been shown, the number of activated relays is either decreased or increased in event of malfunction in one relay. This fact may be utilized to design a checking circuit that identifies malfunction. In the present case, the checking circuit should generate an error signal if the number of activated relays does not equal two. It may then be demonstrated that a circuit possessing such check has the quality of self-checking with respect to single faults in the operation of the internal relays.

Let us determine the qualities of systems with self-checking by means of a number of definitions. By analogy with the internal states, we distinguish among the set of output states of a discrete system the set of basic (working) states Z and the set of error (protection) states Z_R. In order to be able to distinguish the sets Z and Z_R, it is necessary to introduce redundancy into the output information.

Figure 3 shows a redundant output circuit for this system. An extra output v_2 has been introduced. In this case, the circuit has four output states. States v_1v_2 = 10 and 01 belong to set Z, while states 00 and 11 belong to set Z_R. The output state v_1v_2 = 10 is generated when the circuit is in internal

state 0101. When the system is in the other three basic internal states, the output generates values of $v_1v_2 = 01$. If the system is in one of the internal error states as a result of malfunctioning of relays Y_1-Y_4, the output generates values of $v_1v_2 = 00$.

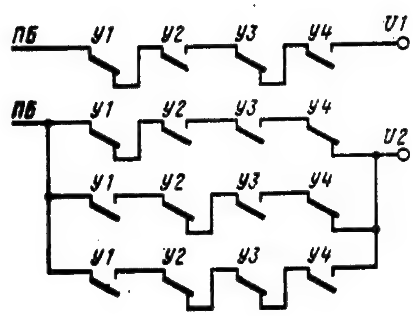


Fig. 3

DEFINITION 1. A discrete system is termed protected against malfunctions if, in the event of any malfunction of a specified class occurring in any input operating sequence, the output states are either correctly computed or belong to a set of protection states Z_R.

DEFINITION 2. A discrete system is termed as self-testing if, for each malfunction of a given class, there exists at least one input working sequence in which at least one output state appears belonging to the set of protection states Z_R.

The quality of being protected against malfunction prevents incorrect actions on the part of the discrete control system on the object under control (the quality of safety). The quality of self-testability eliminates the presence in the system of undetectable malfunctions of a given class and prevents the possibility of their building up. The operating inputs at the same time comprise the checking test. Thus, a system meeting the conditions of definitions 1 and 2 in the indicated sense will check itself.

DEFINITION 3. A discrete system is termed fully self-checking or fully self-testing if it is protected against malfunctions and is self-testable.

The circuit shown in Fig. 2 and 3 possesses the indicated qualities with respect to single malfunctions of relays Y_1-Y_4, since any malfunctioning of a relay will switch the circuit into one of the error states, corresponding to an output protection state of $v_1v_2 = 00$.

The above definitions are also fully applicable to microprocessor or systems, which belong to the class of discrete systems. Since a microprocess system represents a harmony of hardware and software, it is also advisable to design hardware and programs with self-checking.

Design of Hardware with Self-Checking

The key point of design of hardware (a finite-state automaton) with self-checking is as follows. When a malfunction of a given class occurs, the

automaton is switched from the set of basic internal states S to the set of error states S_R. The reverse transition is prevented for all operational input sequences. Protectedness and self-testability are ensured by the fact that output protection states from Z_R are assigned to all states from S_R. The check may be done with respect to either an internal or output state of the automaton. The task of the checking system is to register the transition of the automaton from set S to set S_R, i.e., in the final analysis, to distinguish the basic and the error internal or output states from each other. This task is handled rather easily if the internal state from set S or the output states from set Z are encoded with an error detection code.

Figure 4 shows the structure of a finite-state automaton with self-checking. It contains an input converter BXП, generating groups $a_1, \ldots, a_r$ of input variables $x_1, \ldots, x_p$; a logic converter ЛП, which evaluates the functions of activation of the internal memory elements $y_1(t), \ldots, y_k(t)$; a memory unit БП, containing memory elements; an output converter BП, which evaluates the output functions $v_1, \ldots, v_q$. The checking circuit KC may be connected either to the outputs of БП (self-checking by an internal state) or to the outputs of BП (self-checking by an output state).

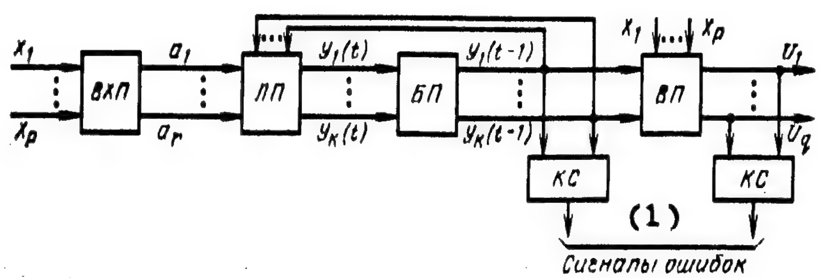


Fig. 4

Key:

1. Error signals

Let us consider the requirements placed on the layout of an automaton with self-checking. Five conditions must be fulfilled in order to effect self-checking by an internal state.

CONDITION 1. Whenever a malfunction q of a particular class occurs, the automaton should switch to a state in the set S_r. The fulfillment of this condition has already been illustrated (see Fig. 2). The class of malfunctions in question is comprised of malfunctions of type $1 \rightarrow 0$ and $0 \rightarrow 1$ of relays Y_1-Y_4.

CONDITION 2. If a malfunction q occurs in a certain operating cycle of the automaton, then in this and all subsequent cycles the automation should operate either in accordance with a specified diagram of transitions (we shall say that the diagram of transitions specifies the function of transitions of the automaton) or, in the opposite case (i.e., if the function of transitions is disturbed), it should switch to a state in the set S_R. This requirement lets us forego ascertaining the malfunction at the instant of its occurrence, if at this time it does not affect the proper functioning of the automaton.

For example, if the circuit shown in Fig. 2 is in the initial state 0110 and a break occurs in the winding of relay Y_4, this malfunction does not affect the running of the system at the given instant or in the following cycle (when the transition 0110 → 1010 occurs), and will only have an influence in the second cycle during transition 1010 → 1001 (at which time it should be registered).

CONDITION 3. The transition mentioned in condition 2 should occur no later than the first cycle in which the output function is disturbed. The output function establishes a correspondence between the internal states of the system and the values of the outputs. The fulfillment of the requirement is necessary in the case when the values of the outputs are determined not only by the internal state, but also by the values of the inputs. Different output values may correspond to a single internal state. Situations may occur when the output function is disturbed under a certain malfunction, but the transition function is not affected. There is no such case for our example.

The above three conditions determine the behavior of a properly functioning and a malfunctioning automaton in the basic states. But in addition to this, it is necessary to determine the behavior of the automaton in the error states, since in accordance with the aforesaid property of an automaton with self-checking after a transition to the set of error states S_R the reverse transition to the set of basic states S is not permitted.

We shall designate by $S_q (S_q \subset S_R)$ the set of such error states into which the automaton may pass if a malfunction q has arisen in it. We distinguish two kinds of malfunction: catastrophic failure and soft failure. In a catastrophic failure, the logic structure of the automaton is changed. The transition function of the automaton will also change, and certain internal states may be impossible for the automaton to achieve. If a soft failure (restorable failure) occurs in an element of the automaton, the logical network of the automaton is restored after disappearance of the failure, in the same way as its transition function.

The following two conditions should be granted with respect to error states.

CONDITION 4. When a malfunction q occurs, the transition function of the malfunctioning automaton should specify a transition from all error states S_q to one of the error states.

CONDITION 5. If the malfunction q is a soft failure, the transition function of a properly functioning automaton should specify a transition from all error states S_q to one of the error states.

These two requirements are fulfilled for the system shown in Fig. 2. Furthermore, in a catastrophic failure and a soft failure in the operation of relays Y_1-Y_4, the system during the cycle of occurrence of the fault or in the following operating cycle will pass either into state 0000 (faults of type 1 → 0) or state 1111 (faults of type 0 → 1). The fact of the transition of the system into these states may be used to design a checking circuit.

Since the system in Fig. 2 corresponds to all of the five above conditions, it possesses the quality of self-checking by an internal state. All internal

error states correspond to an output value of $v = 0$, which in the present example may be regarded as a protection state.

Two further conditions are added to the above five when organizing a self-checking by output state.

CONDITION 6. When a malfunction q is present, the output function of a faulty automaton specifies a protection output value for each state S_q.

CONDITION 7. If the malfunction is a soft failure, the output function of a properly working automaton specifies a protection output value for each state.

These requirements determine the structure of the output circuits of the device. For example, if the structure in Fig. 2 is supplemented with output circuits shown in Fig. 3, the resulting system will possess the quality of self-checking by an output state. In this case, the detection of faults does not require construction of a checking circuit that ascertains the transition of the system from the set of states S to the set of states S_R: the check may be done by comparing the signal values at outputs v_1 and v_2.

The primary method by which the above requirements are satisfied involves encoding the states of the automaton by an error detection code and a particular design of the logic converters which realize the functions of the transitions and the outputs of the automaton. Methods have been developed using balance codes, parity check codes, repetition codes, and summation codes. Such codes possess certain distinguishing features and the layout of the automaton is built so that any malfunction in a particular class results in violation of these features. The class of failures usually consists of all single malfunctions, these being the most likely.

An important element of the structure of an automaton with self-checking is the checking circuit (see Fig. 4), since its reliability largely governs the reliability of the automaton. This determines whether a code vector arriving at the inputs belongs to the given code. A self-testing checking circuit is built in the form of a device having n inputs and two outputs z_1 and z_2 (Fig. 5). It should possess two properties: the property of checking the input vector--outputs z_1 and z_2 take on values (1, 0) or (0, 1) if a vector of the given code is present at the input of the properly functioning KC, and values of (0, 0) or (1, 1) if a vector not belonging to the given code is applied to the input; and the property of self-testing--for each single malfunction of the KC there exists an input vector of the given code at which the outputs z_1 and z_2 assume values of (0, 0) or (1, 1). Figure 6 shows a self-testing KC for the "2 out of 4" balance code.

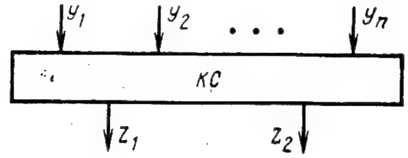


Fig. 5

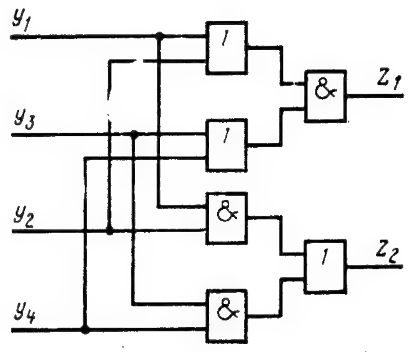
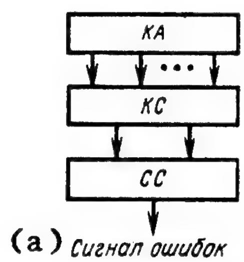


Fig. 6

The general structure of an automaton with self-checking, employing a self-testing KC with two outputs, is shown in Fig. 7. For any single malfunction in the structure of the automaton or in the KC itself, identical signals z_1 and z_2 are set up on the outputs of the latter. This is registered by a totally reliable coincidence circuit (CC) of signals z_1 and z_2, which generates an error signal and acts on the device, disconnecting the faulty automaton from the object under control. It is also possible to use single-output self-testing KCs.



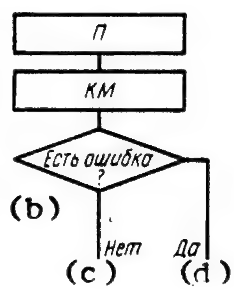


Fig. 7 Fig. 8

Key:

a.	Error signal	c.	No
c.	Error?	d.	Yes

Design of Software with Self-Checking

As we know, the software is much "richer" than the hardware in consequences of malfunctions. Therefore, the question of design of self-testing control programs is also more complicated. Yet the general concepts remain valid. Let us define the concept of a program with self-checking. We shall assume that the program itself has been written properly (without mistakes) and it is necessary to provide protection for the microprocessor control system in running the program against hardware defects. Of the total set of program data, we shall distinguish the set of basic (working) values W and the set of error (protection) values W_R.

DEFINITION 4. A control program is said to be protected if, upon occurrence of malfunctions of hardware in a particular class during any given working sequence of input data, the output data are either properly computed or they belong to a set of protection values W_R.

DEFINITION 5. A control program is said to be self-testing if, for each malfunction of the hardware in a given class, there exists at least one working sequence of input data in which at least one protection value of the output data from the set W_R occurs.

DEFINITION 6. A control program is said to be fully self-checking or fully self-testing if it is protected against malfunctions and is self-testing.

The general structure of a program with self-checking is shown in Fig. 8. Like the automaton with self-checking and its checking circuit, the self-checking program Π has a checking module KM. The task of this is to analyze the output program data and decide whether an error has occurred in the computation. The checking module itself should have self-checking, i.e., it should detect its own errors.

We introduce the concept of the state of a microprocessor system u as the contents of the internal registers of the microprocessor. For a given control program, we may distinguish the set of working states of the microprocessor system $U = \{u_1, u_2, \ldots, u_k\}$ and the set of working input data $I = \{i_1, i_2, \ldots, i_n\}$. The control program realizes a certain algorithm (function) Ψ:

$$W = \Psi(J, U).$$

The function Ψ defines, in deterministic manner, the output program data w in dependence on the input data i and the state of the microprocessor u at the instant of arrival of these input data.

We shall formulate the two main requirements on self-testing programs. We shall assume that the program defects involve a substitution of the object $(c_i \rightarrow c_j)$, functional $(f_i \rightarrow f_j)$, and predicate $(p_i \rightarrow p_j)$ characters.

FIRST REQUIREMENT. Defects causing substitution of object characters distort the input data. This distortion should translate the input data into the domain of error values I_R, and the algorithm Ψ should compute a protection result for these values.

SECOND REQUIREMENT. Defects resulting in substitution of functional or predicate characters either distort the function Ψ, converting it into an error function Ψ', or distort the internal state of the microprocessor. In the former case, the function Ψ' should produce a correct or protection result. In the second case, the distortion of the internal state either should not disturb the computation process or it should switch the microprocessor to the set of error states U_R, relative to which the algorithm produces a protection result.

Organization of a General Self-Testing Checking Circuit for Microprocessor Systems

In order to implement complex control algorithms in the design of discrete systems, the methods of block synthesis are usually employed. In this case, the system is realized in the form of individual, interconnected, independent blocks. These blocks are identified by analysis of the functional algorithm of the system. When microprocessors are used, microprocessor systems are created in this way.

In organizing the self-checking, each block of such system is built as a self-testing device and provided with its own checking circuit (see Fig. 9), which is realized as a self-testing circuit with two outputs. The problem arises of designing a general self-testing checking circuit that unifies all blocks of the system. The idea of the solution of this problem is shown in Fig. 9. The system consists of three blocks, each of which has its own KC. The outputs of two KCs are fed to the inputs of a "2 out of 4" code KC. When the device is running properly, the inputs of 2/4 KC (see Fig. 6) receive a set of groupings {0101, 1001, 0110, 1010}, which constitute a verification test of the given circuit. Thus, all KCs are combined into a common checking circuit by the 2/4 KC cascade connection.

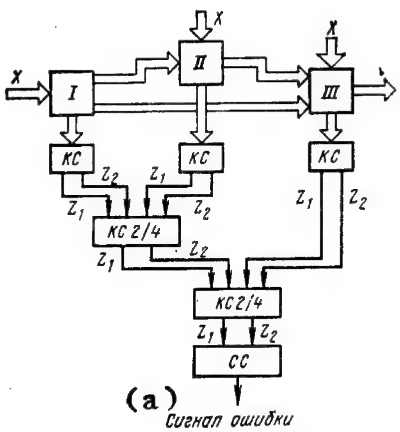


Fig. 9

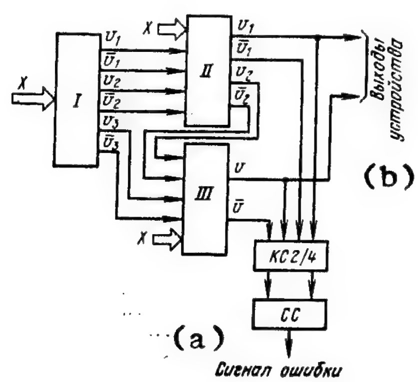


Fig. 10

Key:

a. Error signal
b. Outputs of device

In order to reduce the number of checking circuits in the block system, it is possible to use the principle of checking the operation of one block by another block. This enables total monitoring of the system by means of checking circuits that are connected only to those outputs of the blocks that are directly associated with the outputs of the entire device.

Figure 10 shows an example of the organization of the checking of a system consisting of three blocks. Each block is built as a self-testing device with paraphase outputs. The paraphase outputs of block I are sent to the paraphase inputs of blocks II and III. The checking circuit is hooked up to the paraphase outputs of blocks II and III, the direct values of which are sent to the external outputs of the device. If a malfunction arises within a

particular block, it will result in identical signals ($U_i = v_i$) in at least one of its paraphase outputs, which is equivalent to a malfunctioning of the paraphase input of the block connected to the faulty block. As a result, identical signals are also established in at least one of the paraphase outputs of the second block, which has a similar effect on the operation of the following block, and so on. In the final analysis, identical signals are established in at least one of the paraphase outputs connected to the external outputs of the device, and this is registered by the self-testing check circuit. For example, say a malfunction arises in block I, as a result of which the relationship $v_1 = \overline{v}_1$ is established. This is perceived by block II as a failure in its input converter. At the output of block II, either the relationship $v_1 = \overline{v}_1$ is established, which is registered by the check circuit, or the relationship $v_2 = \overline{v}_2$ is established, being perceived by block III as a failure in its input converter. The relationship $v_1 = \overline{v}_1$ is established at the output of block III, and is registered by the KC.

Simultaneous use of both principles may be effective in certain cases.

UDC 681.3

Software for an Urban Development Model

907G0044A Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 4, Oct-Dec 89 pp 24-27 [MS Received 28 Mar 89]

[Translation of article by I. I. Brona]

[Text] Urban development modeling based on a general plan is one aspect of efforts to develop an automated metropolitan management system. This process is characterized by a the following features:

- large-scale tasks;
- a substantial role of the human factor in both the management system and the managed facilities;
- "fuzziness" and a distributed nature of the managed facility;
- a close relationship between the management levels of real-time processes and solutions to long-term problems (future development of the territorial system);
- predominance of coordination activities of divisional (departmental) services within the managed territory;
- significance of the procedures in preplanning solution preparation, etc.

The general urban development management model must be based on actual processes and account for the need to integrate existing individual "automated islands" into management automation in order to obtain results in the form of automated systems designed for providing information for urban development decision making. The urban development model requires a substantial time period, highly-qualified personnel, new, nonstandard programs and software.

An analysis of the development and implementation of the general urban plan demonstrates that it is important to automate a number of procedures which include the following:

- support of data bases containing information on the present urban condition (on urban construction facilities, services, housing availability, engineering and communications networks, land, population);
- analysis of the urban condition taking scientific and technical achievements in urban construction and industry into account;
- calculation (forecasting) of the demand for urban buildings for the design period accounting for given service standards and natural resources;

• calculation of the pollution, adult and family structure of the municipal populations;

• prediction of the state and demand for the primary resources over the calculated period (labor resources, transportation, communications and engineering networks and energy resources) as well as the state of the environment and land;

• modeling of decisions implemented to eliminate the disbalance between the availability and demand for resources and the consequences of such solutions;

• modeling of decision making for implementing municipal development processes.

Resource planning.

In this subject domain which will be referred to as "municipal development" the following function as the primary aspects: territorial regions, communications-engineering networks, urban construction facilities, service facilities, municipal population, labor resources, housing and environment.

The state of each of these objects will be reflected in the local data base. Then the state of the developing urban area is represented as a vector whose components represent the states of the objects outlined above. It is, however, important to account for the fact that the state of each object also represents the value of a certain complex data structure which will be calculated based on the state of the local data base corresponding to the object and based on the links between this object and other objects in the subject domain.

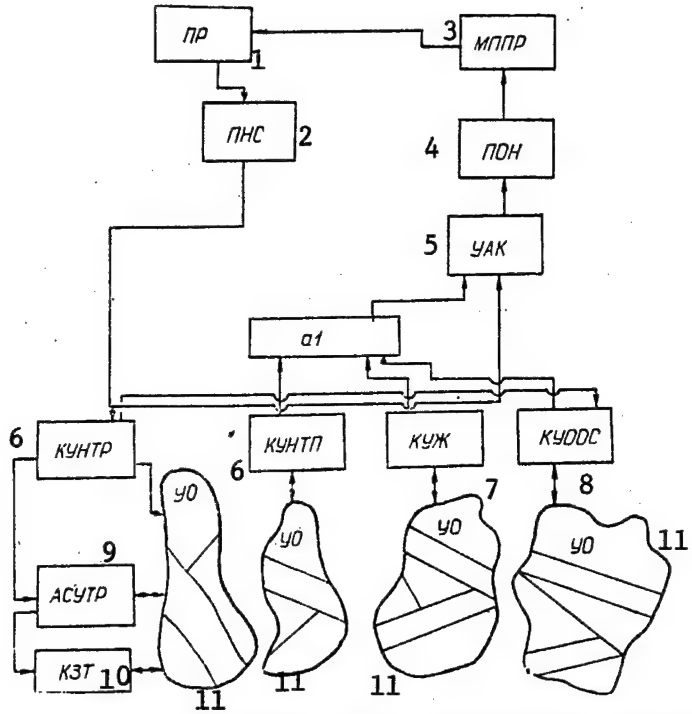
The urban development management system can be represented as consisting of two types of systems: interdivisional systems, multidivisional systems, and functional systems.

The structure of one of the multidivisional management systems entitled *pollution and labor resources* is shown in the figure. For example, this includes the "labor resource management" automated system which in turn contains the "labor management" problem set. Each hierarchical level has a set of problems, an automated system, a management system, a functional system, and data bases. Data bases common to several systems may also exist.

There can be three states of a territorial system at any time: the present state *a1*; the planned state *a2*, and the goal state *a3*.

The multidivisional and interdivisional systems form the representation of the present state. In simplified form each consists of a single state vector component.

The planned state is defined by preparing a socioeconomic municipal development plan. The present state is fixed by the departmental, divisional, and interdivisional systems. Each of the states (even with a rather detailed representation) is characterized by a comparatively moderate quantity of data. As a rule this includes: the utility of each resource (per person, per cubic kilometer of land, per square meter of living space, etc.), specific cost of services, etc. Detailed and precise data are required to calculate each component of the state; these are provided by groups of automated system tasks in the interdivisional systems.



Overall structure of the multidivisional automated municipal development management system.

1 - Decision making system; 2 - New state planning system; 3 - Decision making consequence modeling system; 4 - Plan substantiation of new state; 5 - Accounting, analysis, testing; 6 - Scientific and technical progress; 7 - Domestic housing; 8 - Environmental protection; 9 - Labor resource management; 10 - Labor management; 11 - Managed object

The goal state is defined by preparing long-range solutions. One such solution is to adopt a general municipal development plan.

Functionally the problems that are assigned to the upper level involve selecting coordinated actions for multidivisional systems so that the management processes implemented by the systems will help to achieve the goal state as quickly as possible. The following functional systems are provided for this purpose on the upper level: decision making, new state planning, decision making consequence modeling, planned substantiation of new state, accounting analysis and testing; the following functional systems are on the lower level: interdivisional and multidivisional population and labor resource management, scientific and technical progress management, domestic housing management, environmental management, and managed object.

The decision making block for planned state implementation will be implemented as an expert system in the future. The interdepartmental and interdivisional systems will be subjected to actions from the upper level with respect to the planned distribution of resources as well as plan decisions to maintain the new intermediate states of these systems.

Two features that complicate the use of classical coordination methods for urban development planning should be noted. First the decision making process is not implemented as algorithms and largely is designed to account for the human factor.

Second the involvement of a large number of facilities in managing individual systems in the expanding municipal area cannot be divided into independent sections without disrupting the relationships.

An automated decision making system with an advanced expert assessment unit for assessing solution versions represents the most effective method of accounting for the human factor. Regarding the disruption of relations between individual departments this problem has not yet been investigated in the fundamental plan. Relation significance criteria, availability periods and effects are not available.

Research is currently being conducted to substantiate methods and means for solving the problems outlined above. Special attention is currently focused on an investigation of an information-logic municipal model and an implementation of this model using a data base system. All municipal objects (land, communications engineering networks, domestic housing, multipurpose departments, etc.) have a special data base which is called a register. The information-logic municipal model reflects the relations between these objects as well as certain auxiliary objects which are specially introduced.

It is possible to implement the information-logic municipal model in several versions. One version has been developed that employs a data base integration system [1]. The integration system is used for generating requests on the data base system; each such data base can function as a municipal object register. Data base access can be acquired by request; this data base which is already independent will be used in the future for specific applications. These data bases are arbitrary data bases and relational data base management systems are used to support them.

A graphics data base which takes the form of a municipal map or a map section is used to display information and interact with the user. The computer representation of the municipal map will be used in the future to display modeling results.

A study of the concept of urban develop modeling has made it possible to identify a number of tasks that are best expanded first. It is proposed that they be implemented by means of automated workstations based on personal computers used in local area computer networks or independently. The information interaction processes between the automated workstations should be handled first; information interaction should then be implemented by exchanging local data bases using diskettes.

The functional purpose of automated workstations used in local area computer networks is to determine the planned states of the urban environment en route to achievement of the goal state together with the required resources, balance the rates of growth of the latter with shortages so that the goal-state is achieved at the scheduled time.

Independent automated workstations are designed for social infrastructure facilities (service facilities) and their resources in order to support a balance of services in the local urban regions [2].

Automated workstations are also developed for other applications.

The implementation of the urban development modeling process by means of automated workstations involves both software and data base support of the urban

development model. It is recommended that an information model that represents the current state of the urban environment based on integrated factors be developed. Such a model can be expanded in two ways: by calculating the integrated factors of the municipal facilities using the systems data bases and producing expert conclusions on the state of the urban facilities.

This information system and its individual components are designed for municipal planning and economic services and design organizations responsible for the implementation of the general urban plan. The system is currently undergoing experimental testing in Kiev.

References

1. Olenin, M. V. "Integration of data bases for information support of regional management system", Metody i sredstva razrabotki informatsionnogo obespecheniya dlya zadach upravleniya, Kiev,

2. Brona, I. I., Aleksandrenko, V. L., Dyumin, N. V., "Hardware and software for local regional analysis and its application for optimization of service facilities" Tez. dokl. Vsesoyuz. konf. "Territorial'nye neodnorodnye IVS" [Conf. Proc. of the All-Union Conf. "Territorial Inhomogeneous Data Processing Networks"], Novosibirsk, 2-4 Nov. 1988.

3. Demin, N. M., Shuster, A. A., "General metropolitan development plans and software", Goroda i sistemorasselenie, Moscow, 1984.

UDC 681.52

A Microprocessor System for Automation of Raman Scattering Experiments

907G0156A Dushanbe DOKLADY AKADEMII NAUK TADZHIKSKOY SSR in Russian Vol 32, No 8, Oct 89 (manuscript received 13 Jan 89), pp 514-516

[Article by T. Kh. Nazarov, R. M. Asimov, A. B. Ilyayev and N. S. Abdulloyev, Physical-Technical Institute imeni S. U. Umarov, Academy of Sciences, Tadzhik SSR: "A Microprocessor System for Automation of Raman Scattering Experiments," presented by A. A. Adkhamov, member of the Academy of Sciences of the Tadzhik SSR, 25 Nov 88, published under the rubric "Computer Science"]

[Text] Laser spectroscopy is widely used in studies of crystal characteristics. Raman light scattering [KRS] is one such method. The spectrum of the light scattered by a crystal conveys important information on its properties, such as dispersion of dielectric permittivity, refractive indices, electrooptic characteristics, etc. It is essential that experiments working in this field obtain a quality scattered light signal.

An experimental installation for KRS can measure the wavelength of scattered light and the corresponding intensity. Among the chief difficulties experimenters encounter are isolating the valid signal from the background noise and rapid processing of the flow of incoming data. An automated microprocessor system has been designed and built for overcoming these difficulties; it accumulates data received during the course of an experiment, processes these data, stores and outputs data to a plotter or cathode-ray tube [ELT]. The system is implemented on the basis of an Elektronika D3-28 microcomputer.

The system differs from existing models [1, 2] in that it is of a much smaller size and is less expensive and more easily available (being based on a widely available Soviet-made microcomputer); it is reliable due to a simplified interface design and digital communication with experimental equipment.

Figure 1 shows the flowchart of the automated system. Here, 1 is the experimental equipment (DFS-24, laser and specimen to be investigated), 2 is the

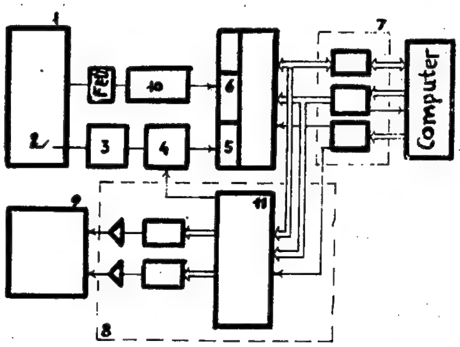


Figure 1. Flowchart of the microprocessor system for Raman spectroscopic investigations.

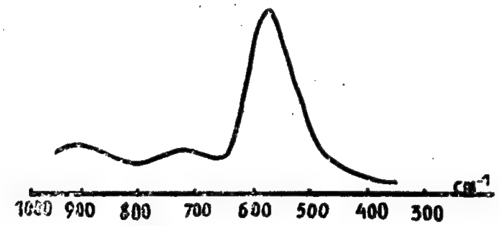


Figure 2. Raman spectrum of an LiNbO crystal (scattering geometry Z(YZ)X) generated by the microprocessor system.

wavelength control sensor, 3 is a flicker suppression circuit, 4 is an initialization pulse-forming circuit, 5 is the current wavelength counter, 6 is a photon counter, 7 is the bus driver, 8 is a unit for output of data from the computer to a plotter, 9 is the plotter (or ELT), 10 is a radiometer, 11 is a programmable adapter and 12 is a photoelectronic multiplier [FEU].

An electromechanical device functions as the wavelength control sensor; it consists of a four-vane impeller with magnetic couples (magnet-reed relay) and is linked mechanically to a diffraction grating drive.

The contact flicker suppression circuit is based on an RS trigger. It operates according to the principle of consecutive registration of signals from two reed relay contacts placed at a distance from each other. The distance between the contacts is determined by the design of the wavelength control sensor and should be large enough to preclude possible simultaneous triggering of the contacts.

The initialization pulse-forming circuit serves to switch the initializing pulse, software-generated by adapter 11 and the pulses sent by the wavelength control sensor and the flicker suppression circuits.

Counters *5* and *6* are built on the basis of a programmable KR580VI53 interval timer.

The photon counter is implemented on the basis an FEU-79, Robotron-20046 radiometer and counter *6*. The circuit for output of spectra to a plotter or an ELT consists of a programmable adapter, two KR572PA1 digital-to-analog converters and current-voltage converters.

The system is connected to a microcomputer through the bus driver 7, which ensures electric and logical compatibility with the microcomputer. The software consists of input and output programs and programs for accumulation and smoothing of input spectral data and primary spectrum processing.

Initial and final wavelengths in the region of scattered light being registered are fixed as the main parameters of the input program. A discrete variation of the scanning step in terms of wavelength makes it possible to investigate spectra in any frequency region.

Preliminary spectrum-processing programs recalculate the spectrum in terms of wavelength into a frequency spectrum and determine frequencies, half-widths and integral intensities in the spectra.

Figure 2 is an illustration of a KRS spectrum for a nonlinear LiNbO crystal in the frequency range 10-380 cm (scattering geometry Z(YZ)X) produced by the system.

The system substantially reduces experimental time (by a factor of 8-10) and improves the reliability of the data obtained. In addition, expansion capabilities are provided.

Bibliography

1. Avakyants, L.P., Kitov, I.A., and Chervyakov, A.V., PROBLEMY TEKHNIKI EKSPERIMENTA, No 2, p 145, 1988.

2. Ilyayev, A.B., Shefer, A.D., and Dubovik, M.F., in: TEZISY DOKLADOV III VSESOYUZNOY KONFERENTSII PO SPEKTROSKOPII KRS [Abstracts of Papers Presented at the Third All-Union Conference on Raman Spectroscopy], p. 186, 1986.

Automation of Experimental Studies of Signal-Noise Radar Conditions

907G0163A Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 3, Mar 1990 (manuscript received 13 Sep 1989, after revision 7 Oct 1989) pp 118-128

[Article by A. A. Ursatyev, I. G. Prokopenko, S. L. Sapozhnikova, S. A. Tarasenko, Yu. M. Greshishchev

[Text] Despite substantial experimental material on the energy and fluctuation characteristics of echo signals, for example, in marine radar [1-3], there is still a need for in-situ studies of the signal-noise conditions. The complex interaction of the object of radar observation and the surrounding medium, as well as a variety of phenomena and conditions of detection, predetermine the statistical nature of radar reflections. The mechanism of radio wave scattering has been insufficiently studied, and this causes significant difficulties in the creation of mathematical models of the processes that are occurring, and makes experimental studies to solve a number of practical and fundamental problems prominent.

In marine radar where the agitated sea surface is an underlying surface in the detection of radar targets, reflections from the surface form passive noise which masks the target. The creation of mathematical models of echo signals constructed on the basis of the theory of the scattering of radio waves by objects and the sea surface is made extremely difficult by the fact that the sea surface,as well as underwater objects, are anisotropic and have a complex structure which varies over time. This structure can be characterized only be statistical methods.

Moreover, one cannot consider the complex interaction of the atmosphere-ocean system and its effect, nor the effects of other specific reflectors (for example, biological objects) on the characteristics of the signal reflected by the radar target, or the dependence of the reflection conditions on hydrometeorological processes. In [3] it is noted that the fundamental features of the ocean-atmosphere system and the scales of atmospheric and oceanic movements leads to scattering, and sometimes contradictory results.

Thus, various models of echo signals are presently used in the processing of radar information, from very simple models (one-dimensional, Rayleigh distribution) to multiply connected Markov processes with nongaussian one-dimensional distribution, which, however, do not exhaust the variety of situations arising in the course of using radar devices [3, 4]. Most processing algorithms have been obtained for idealized mathematical models of useful signals and interfering background reflections [2, 4].

Thus, the solution of new problems, aside from a priori information on the characteristics of reflections and an understanding of the physics of the processes which are occurring, requires in-situ studies. These studies consider the entire set of factors which cause fluctuations of echo signals. As a result, this is the main method of determining the statistical characteristics of radar reflections. Among the new problems is the radar detection of a new class of targets (with the architecture of the target and its effective area of scattering, etc., unknown beforehand) and in other wave ranges; radar stations now being planned, as well as systems of automatic radar course plotting; and the evaluation of their effectiveness. There are also a number of other problems requiring a study of the objects of reflection and characteristics of the reflected signals.

In-situ studies can be efficient only when the processes of collecting and processing data are fully automated. In order to obtain objective data on the statistical characteristics (correlation functions, spectra, and probability distribution laws) of echo signals produced in different ways, and to verify the suitability of statistical models of the signal-noise situation to the real characteristics of signals, a complex of hardware and software was developed which is oriented toward conducting in-situ studies with pulse noncoherent radar stations. Since only the statistical properties of rounding processes make sense for noncoherent radar stations, the signal is collected from the output of the video detector. To solve problems which require acquisition of the signal characteristics, which are invariant to the type of radar station used, the system may correct (using a program) the recorded data, considering the distortion of signals introduced by the receiving channel of the radar station.

A system to automate experimental studies of the characteristics of signal-noise radar conditions (hereinafter, the system) was created on the basis of a one-board microcomputer (MS 1201.01, MS 1201.02) and special units (radar station video signal coder and interface) which provide a functional orientation toward work with noncoherent pulse radar stations. The system may also use older models of professional personal computers in the DVK series [5] or other microcomputers of the following families: Elektronika-60, Elektronika MS 1211, or Elektronika MS 1212 [6, 7].

The Elektronika-60 computer was chosen to be the basic microcomputer of the system. The Elektronika-60 has an M2 central processor, which made it possible to use a number of developments in the area of general system and applied software. The system (Figure 1) used the Elektronika MS 7401 graphic display.

Special system units were used in the construction of the Elektronika-60 microcomputer and were installed in free positions of the MPI or MI2 bus [8]. The coder is the video signal digital coding device, which is based on super-fast integrated circuits (analog to digital converter K1107 PV1 (PV2)) [9] and performs 6(8)-bit analog to digital conversion of the signal in a frequency band up to 7 MHz wide with a quantization $T_{ti} \geq 50$ ns. The coder uses an active scheme to recover the constant component of the video signal in the interval coupling the "black" video signal level when the radar station receiver is closed for the duration of the sounding pulse.

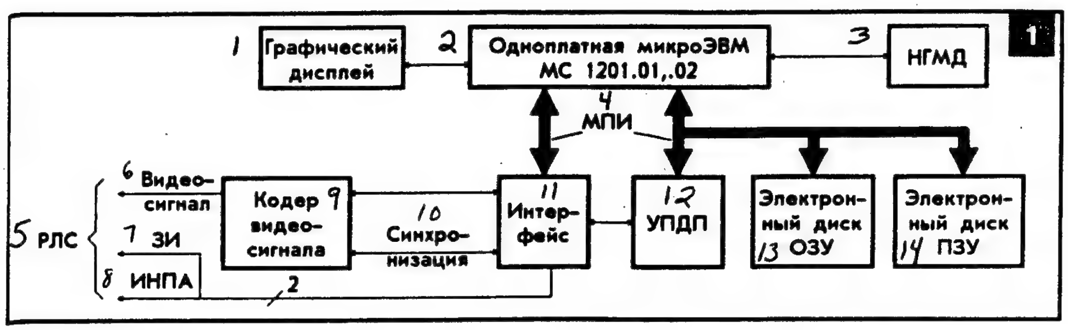


Figure 1. Functional schematic of the system to automate experimental studies. 1. graphic display; 2. one-board microcomputer, MS 1201.01, .02; 3. floppy disk drive; 4. MPI bus; 5. radar station; 6. video signal; 7. start-up pulse; 8. pulse of the null position of the antenna; 9. video signal coder; 10. synchronization; 11. interface; 12. UPDP [not further expanded]; 13. electronic RAM disk; 14. electronic ROM disk.

The interface[1] matches the temporal characteristics of the microcomputer channel, which is relatively slow, with a high information conversion rate. Moreover, this unit of the system is used to adjust to the required operation modes, implementing the spatial and temporal parameters for the collection of data specified by the user. It also synchronizes the operation of the radar station with the system and:

—assigns the data collection mode for radar measurements (in a strobe, in circular survey mode, and in the standing antenna mode of the radar station);

—forms the strobe coordinates in the survey zone of the radar station from initial data: the bearings of the beginning (B_b) and end (B_e) of the strobe and the distance of the beginning (D_b) and end (D_e) of the strobe;

—establishes the quantization frequency F_{ti} of the video signal in the spatial coordinate (distance to the object);

—establishes the recording period of data in the temporal coordinate with frequency F_{su} of the start-up of the radar station (the frequency of the packet of sounding pulses) or with frequency F_{su}/K of the start-up pulses reduced by the required factor of k ($k \leq 2^{16}$);

—forms the address of the scratch-pad memory according to the strobe length in the spatial coordinate $D_e - D_b$ and the quantization frequency of the video signal F_{ti} (the maximum number of resolution elements in the spatial coordinate M_d max = 1024);

—corrects the null distance D_0 of the system strobe with the null distance of the radar station;

—chooses the discreteness ΔD_0 of the distance measurement of the strobe depending on the established distance scale of the radar station (in large scales (1 or 2 miles) and in one of the medium scales (4 miles), with a possible value of ΔD_0 = 0.008 miles)[2];

—forms an indication of the type of synchronization system, self-synchronization or synchronization from the radar station (MASTER or SLAVE mode) and generates

[1]Yu. L. Serebryanikov, junior research associate, participated in the development of the interface.

[2]The nautical mile is equal to 1852 m, and a cable length is 185.2 m.

a synchronization sequence to start up the radar station;
—boots the system using a program or an external start-up source.

The exchange of information between the central processor and the interface is done by program operations using program interrupts, and in direct computer memory access mode. The interface appears to the central processor in the form of ten addressable registers:

1) 16-bit system state register
2) 4-bit system control register
3) counter for the bearing at the beginning of the strobe
4) counter for the bearing of the end of the strobe
5) 9-bit register—counter for the distance at the beginning of the strobe
6) 10-bit register—counter for the distance at the end of the strobe
7) counter for the frequency of the system start-up
8) counter of start-up pulses
9) register of control word 1
10) register of control word 2.

The last two registers belong to the K58OVI53 LSI circuit of the programmable interval timer [10], in which registers-counters 3, 4, 7, and 8 are implemented.

Combination of the marker of the null strobe distance D_0 with the markers of the null distance of the radar station is done through a program by changing the setting for register-counter 5 to a value proportional to the time delay relative to the beginning of the radiation of the sounding pulse to the beginning of the distance reading in the radar station (BD_0). The content of register-counter 6 defines the number of measurements in the spatial coordinate, the address of the scratch-pad memory. Registers 1 and 2 are needed to assign the modes and control the functioning of the system.

To record settings in the interface registers, a program exchange channel is used. Reference to the registers in the process of system operation is done in response to the need to interrupt a program, which the radar station initiates when the antenna passes the null direction reading. The program to service the interrupt is carried out during the pulse which marks the null position of the antenna. Thus, for each new rotation of the antenna, the parameters of the strobe may be reprogrammed by changing the settings of the appropriate registers.

Settings for the registers are calculated on the basis of data of the interactive dialog of the user with the system in natural language. Thus, the user indicates by what factor the initial period T_{sp} [sp = start pulse] of system start-up must be changed. Register 7 is programmed according to the discreteness of the measurement signal using the bearing given by the user. Registers 3 and 4 are programmed by calculating the number of start-up pulses which determine the degree of bearing at the beginning and end of the strobe

$$\text{BEARING B} = (\text{start pulse of rotation/discreteness of bearing})/360^\circ \times P_b$$
$$\text{BEARING E} = (\text{start pulse of rotation/discreteness of bearing})/360^\circ \times P_e \qquad (1)$$

where the start-up pulse of rotation $= T_a/T_{sp}$ is the number of bearing clock pulses falling within the rotation of the antenna of the radar station with a period of rotation T_a; P_b, P_e in degrees.

Register 5 is programmed by calculating the number of distance quanta which define the beginning of the strobe: number of distance quanta $= D_b/\Delta\ D_0 + BD_0$.

Register 6 is programmed by defining the number of signal measurements in the spatial coordinate of the number of distance quanta. The user, in the dialog, chooses the desired frequency of discreteness F_{tt}. Then

$$\text{number of distance quanta} = 37.04\ \frac{(D_e - D_b)F_{tt} \cdot 10^5}{C} \leq M_d \text{ max}, \tag{2}$$

where C is the speed of light, $3 \cdot 10^5$ km/s; D_b and D_e are in miles, F_{tt} in MHz.

Each measurement is a byte. To increase the rate of exchange between the interface and the RAM of the microcomputer, two bytes of data are stored in the scratch-pad memory of the interface, and these bytes are united into a 16-bit word and transmitted along a direct access channel to the RAM. The direct access channel is based on a standard device, I7 15 KS-16-002, from a number of functional modules of the Elektronika-60 microcomputer [7]. The interface stores the data of measurements of the radar signal of one sounding sample.

Thus, the condition $t_{rd} + t_{sp} < T_{sp}$ (here t_{rd} is the time to record data and t_{dac} is the time to transmit data along the direct memory access channel). The time to transmit one word in batch mode when working with the M2 central processor is about 1.5 μs. For the Elektronika MS 1201.01 microcomputer this time is, on average, 2.7 μs.

The direct access channel start-up pulse, like register 7, is programmed according to the chosen work mode of the subsystem and the dialog data. Thus, the number of words transmitted along the direct access channel is the value of the strobe in words, calculated according to the following expression: strobe in words = (BEARING START − BEARING END) (number of distance quanta)/2. The software verifies whether the memory field of the RAM derived from the set of information satisfies the value of the strobe, that is, whether the following condition is satisfied: strobe value in words $\leq$ RAM memory field. The number of strobes in the set of information is calculated, STR SET = inf (RAM memory field/strobe value in words), and the user is issued the message POSSIBLE TO WRITE TO STROBES.

Information in the external carriers is transmitted to files PASP XX.DAT and KADR XX.DAT. Here XX is the number of the set. The file PASP XX.DAT contains the passport data on the set of information written in the file KADR XX.DAT. The set passport is needed to read information for subsequent processing, and when the system is reproducing radar images.

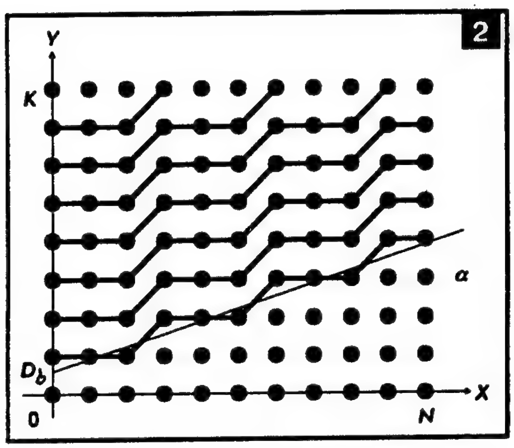


Figure 2. Sequence of data selected for processing. The dots indicate the processed file of data in the coordinate plane $X0Y$; the lines connecting the readings show the data selection sequence for a given processing angle α; D_b is the initial point of the processed file. Processing is subject to an implementation where each volume is no less than N for $|\alpha| < 45°$ and no less than K for $|\alpha| \geq 45°$.

Application program software. The application program software is intended to obtain from the samples a relatively small volume of stable estimates of the statistical characteristics of recorded radar signals [11].

The initial data subject to processing is in the form of a two-dimensional file. The user can study various regular formations from this two-dimensional file and obtain the spatial and temporal characteristics by giving the appropriate rule for the selection of data and by establishing its parameters.

When this file is processed according to a previously defined rule, data is chosen and formed into a one-dimensional file, from which one can obtain all the required statistical characteristics.

Figure 2 shows the sequence of data selection from a file of dimension N, K for a linear rule with parameters α, D_b. The amount of selected one-dimensional realizations is defined by the total volume of processed data N_{tot} given by the user.

The application program software also processes the realizations chosen from the file:
—representation of the rule for selecting data from the two-dimensional file;
—representation of the selected one-dimensional video signal realizations;
—construction of histograms;
—approximation of the empirical distribution with a set of Johnson distributions and estimation of the parameters of the approximating distribution;
—construction of a correlation function;
—construction of an energy spectrum.

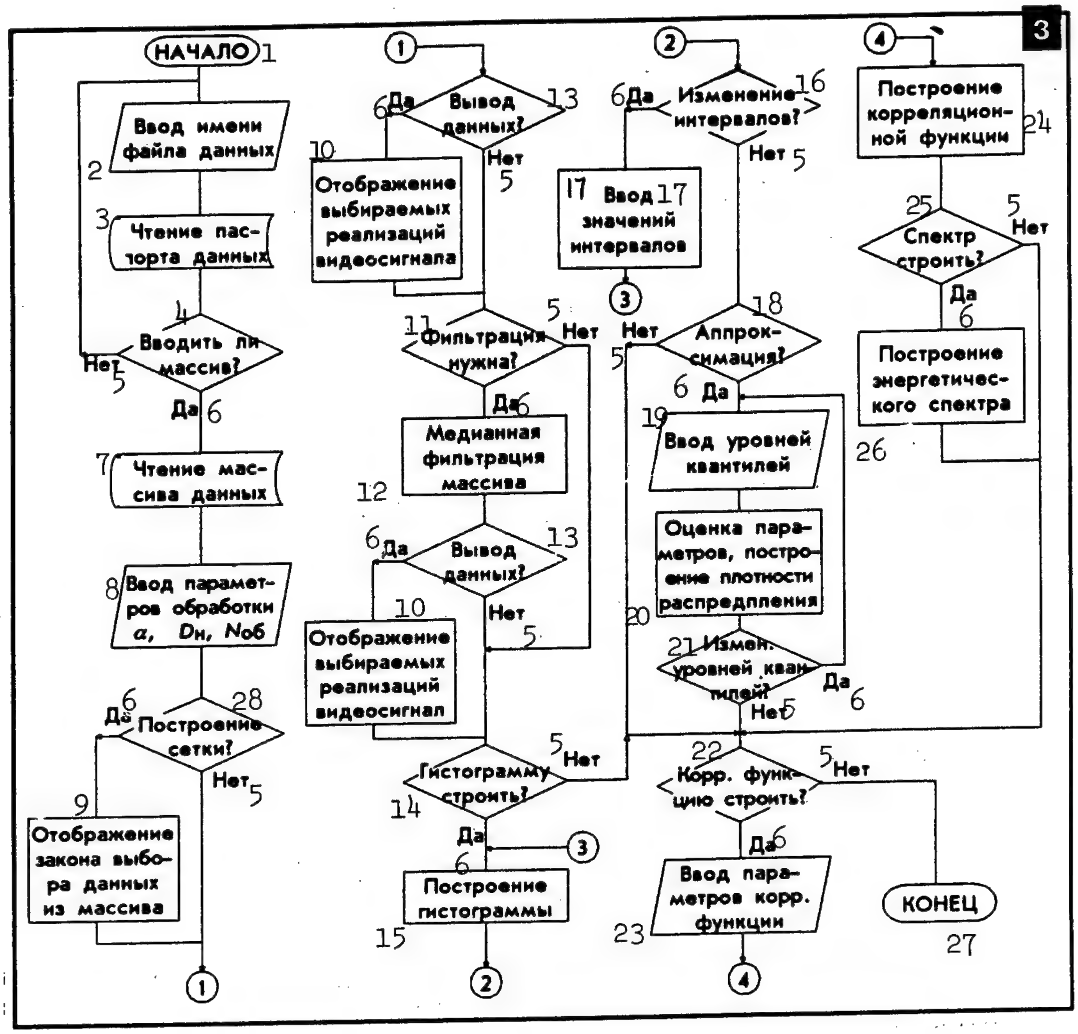


Figure 3. Flowchart of User-Application Software Interaction Algorithm

1. START; 2. Input name of data file; 3. Read data passport; 4. Input file?; 5. No; 6. Yes; 7. Read data file; 8. Input processing parameters α, D_b, N_{tot}; 9. Representation of rule for data selection from file; 10. Representation of the chosen realizations of the video signal; 11. Is filtering necessary?; 12. Median filtering of the file; 13. Output data?; 14. Construct histogram?; 15. Construction of histogram; 16. Change intervals?; 17. Input values of intervals; 18. Approximation?; 19. Input quantile levels; 20. Evaluate parameters, construct distribution density; 21. Change quantile levels?; 22. Construct correlation function?; 23. Input parameters of correlation function; 24. Construction of correlation function; 25. Construct spectrum?; 26. Construction of energy spectrum; 27. END; 28. Construct network?

When the user is in interactive mode, he can choose, using the application program software, the file of data to be processed, establish the parameters of the rule to select data, and implement the required functions for processing one-dimensional realizations of the video signal. The algorithm for user dialog with the application program software is given in Figure 3.

Let us examine some properties of implementation of the application program software functions. In the analysis of the statistical characteristics of the signals it is desirable to examine one-dimensional realizations of the recorded signals, realizations chosen according to a certain law from a two-dimensional file for further processing. Output of the realizations of a random process is done in two ways: in 256-reading segments or the entire realization equally divided into up to 256 readings.

One can examine the realization of a random process before and after filtering of the data file and visually monitor the efficiency of suppression of existing spikes.

The stability of the resultant evaluations protects them from the effect of possible spikes in the processed data, the source of which is the proximity of the radar station, atmospheric noise, disturbances in the equipment's own synchronization, etc. In the application program software, stability is provided by preliminary filtering [12] of the initial data file:

$$\hat{x}_{ij} = \text{med}\,\{x_{i-1j},\, x_{ij},\, x_{i+1j}\}, \qquad i = 1,\ldots,N, \qquad j = 1,\ldots,K,$$

where med $\{\cdot\}$ is the operator to calculate the median of the selection.

In this case, with a filter base equal to three, the median was equal to the second order statistics of the selection.

Median filtering is done on the request of the user, and is done in the direction of the maximum dimension of the two-dimensional file of data ($N > K$). The chosen filter base suppresses single spikes and minimal frequency distortions of the processed signals.

A histogram of the processed data file is first constructed in the finest grid, with intervals corresponding to one quantization level of the video signal. It is then used in the approximation of the distribution function to calculate the selected quantiles. If necessary, the intervals of the histogram may be enlarged (the maximum number of intervals is 256).

To approximate the empirical function of distribution density, a set of Johnson distributions is chosen that describes a rather broad class of distributions of limited random values, including a uniform distribution, U- and I-shaped β distributions, a normal distribution, a γ distribution, an exponential distribution, etc. [13]. The advantage of using a set of Johnson distributions is that the evaluation of the parameters of the approximating distribution may be obtained using the selected quantiles of the empirical distribution.

Approximation of the distribution and an evaluation of its parameters is done using two selected quantiles, the levels of which are given in interactive mode by the user, and which depend on the type of histogram to be obtained. The histogram of the approximating distribution is constructed using the same intervals as the histogram used to determine the suitability of the resultant approximation of the existing histogram.

In calculating the correlation function the user assigns the number and intervals of calculated correlations. The minimum interval is equal to the interval between two adjoining readings, and the maximum interval is limited by the length of the processed one-dimensional realization of the video signal.

Averaging of the calculated correlations is done within the realization

$$K_j(m) = \frac{1}{n-m} \sum_{i=1}^{n-m} \tilde{x}_{ij} \cdot \tilde{x}_{i+m,j},$$

and in several realizations

$$K(m) = \frac{1}{R} \sum_{j=1}^{R} K_j(m),$$

where R is the number of one-dimensional realizations from which the correlation function is calculated.

This property of the averaging algorithm should be considered by the user in the formation of one-dimensional realizations, that is, these realizations should satisfy the requirements of stability and ergodicity.

Calculation of the energy spectrum of a random process is done using a fast Fourier transform of the correlation function which was obtained earlier. One should bear in mind that the estimate of the spectrum defined by the correlation of existing estimates of the correlation function, which, for intervals m long, are close to the length of available realizations n, will be invalid. This must be considered in the definition of the parameters of the calculated correlation function, so that the maximum correlation interval provides a sufficient averaging.

All of the subroutines are written in FORTRAN IV. The general loading module, which was formulated for work under the RT-11 operating system, takes 48 kilobytes of RAM. The maximum size of the file to be processed is $14 \cdot 10^3$ readings.

The software to support the system hardware includes interactive programs, system hardware diagnostics, and programs to adjust the hardware to the given operation mode and carry out the mode. The software occupies 500 words of RAM and operates under any of the RAFOS operating system versions. When the RT-11 operating system is used, the user has 15 kilowords of RAM to store data. The software for real time operation of the system (synchronized with the signal sensor) is in assembler.

There is a 512 kilobyte electronic disk, the external memory of the microcomputer, which is directly connected to the MPI bus (see Figure 1). When the measurement program assigned to the radar station survey is finished (the data of one strobe is written) data is transferred from RAM to the electronic disk. This operation is done within this same sweep controlled by the RAFOS operating system.

Thus, when one electronic disk of several possible disks is reserved for experiment data, and when there is maximal use of the RAM provided to the user for these purposes (15 kilowords) per observation session, one can record (considering the block structure of information records) about 15 stobes 3.2 miles$\times$ 6° in size at a signal discreteness frequency $F_{sp} = 1.5$ kHz. This value of the video signal discreteness frequency provides a resolution in the spatial coordinate $\Delta M_d = 15$ m (0.008 miles or approximately 0.1 cables) and may be determined from expression (2) for one measurement along the spatial coordinate.

If one takes the size of the strobe to be equal to the strobe of the auto-accompaniment of the "Briz-E" SARP [not further expanded] (720$\times$5°) [14], then more than 160 of these strobes may be recorded. In other words, information on radar images may be recorded continuously for 160 rotations of the radar station antenna. As the practice of in-situ studies of reflections from different types of underwater objects using ship-borne radar stations has shown [2, 3, 15], the fluctuations of the echo signals may be viewed as stable for unstable sea waves and backscattering of the object within several dozen antenna rotations.

The energy-independent electronic ROM disk [16] provides high system reliability and efficiency in field conditions due to the fact that it holds the application program software, the software for the hardware, a monitor, and a number of RAFOS utilities. The ROM disk is on a K573RF4 integrated circuit and has a memory of 256 kilobytes. Its operation in the RT-11 operating system is supported by the QD.SYS driver.

However, despite the substantial increase in the productivity of the computing complex, especially in program work, which requires intense exchange with external memory, and despite the increase in the lifetime of the floppy drive and diskettes, the advantage of using a ROM disk in the microcomputer with a small volume of addressable space (up to 64 kilobytes) may not always be fully used. Thus, the inability to use the USR swapping procedure leads to the fact that programs occupying a large amount of memory may not be directly loaded into RAM from the ROM disk. In this case it is necessary to reload the system from the ROM disk to the electronic disk, saving a place for the SWAP.SYS file and then working with the electronic disk as a system device.

Experimental Studies. Several results of the studies of radar reflections were obtained in in-situ testing of the system in a test site, the Karadag Field Experimental Base of the A. I. Voyekov Main Geophysical Observatory.[3] The signal sensor was the ship-borne "Nayada 5" radar station, in the 3 centimeter wave length [17].

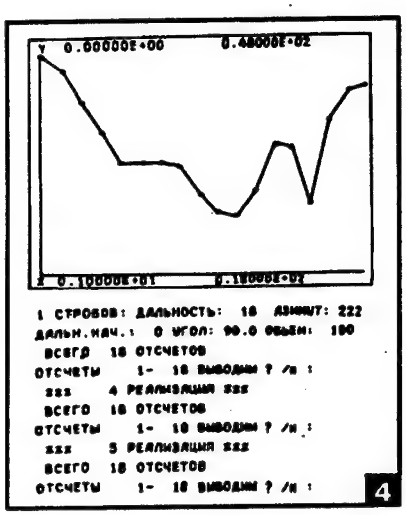
---

[3]The authors express their gratitude to the directors of the A. I. Voyekov Main Geophysical Observatory, and the personnel of the Karadag Field Experimental Base for their active collaboration and help in carrying out the work.

The height of the antenna over sea level was 50 m. Objects of observation were the following: the sea surface, underwater objects (a buoy, beacon, boat, etc), the shoreline, and hydrometeorological formations.

The results of the processing of various observational data are given in Figures 4-7. Figure 4 shows discrete values of a signal reflected from part of a mountainous surface 6 miles away from the radar station (one sounding, 18 elements of resolution in the spatial coordinate). Recording was done with the antenna stopped, and 222 soundings of this portion of the surface were made. In the upper field (see Figure 4) are the minimum (left) and maximum values of the reflected function; the lower field is an increment of the function argument and the maximum value of the function (right); the values of the correlation intervals (in seconds) and the spectrum harmonics (in Hertz) were defined on the basis of information on the collection modes (the set passport) and the processing of the data.

The results of the values of the echo signals from individual points of the observed object, the 18th (last), 16th, and 5th elements of distance resolution (see Figure 4), are presented respectively in each frame of Figures 5-7. In Figure 5, clockwise from the upper left frame are: a fragment of the sequence of values of the echo signal over time, a histogram, and a spectrum constructed from the



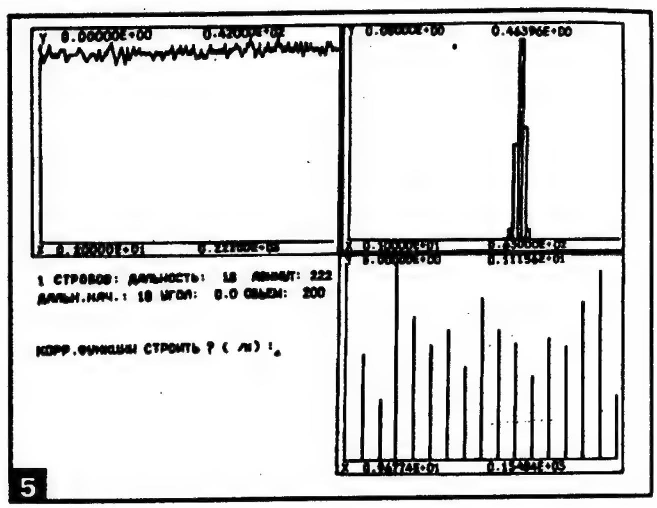
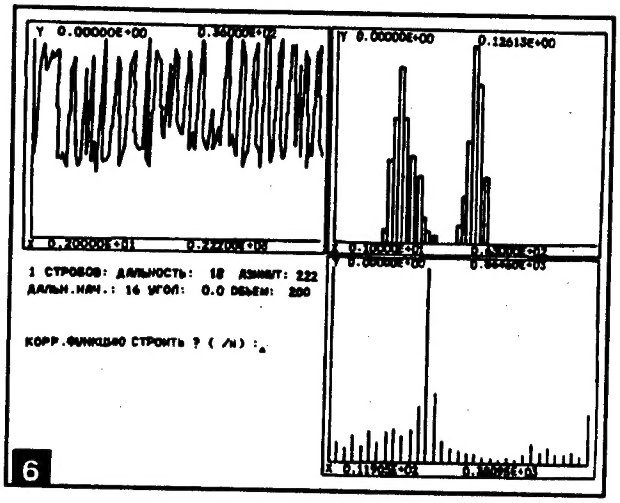


Figure 4 [left]. Discrete values of the echo signal from a portion of a mountainous surface. Information below figure: 1 stobes; distance: 18; azimuth 222; initial distance 0; angle: 90.0; volume: 100; total of 18 readings. Readings 1-18 output? /N: ***4th realization***; total 18 readings; readings 1-18 output?/N:***5th realization***; total 18 readings; readings 1-18 output? /N

Figure 5 [right]. Results of processing of the values of the echo signals from the 18th element of distance resolution. Legend: 1 stobes; distance: 18; azimuth: 222; initial distance: 18; angle: 0.0; volume: 200. Construct correlation function? (/N):



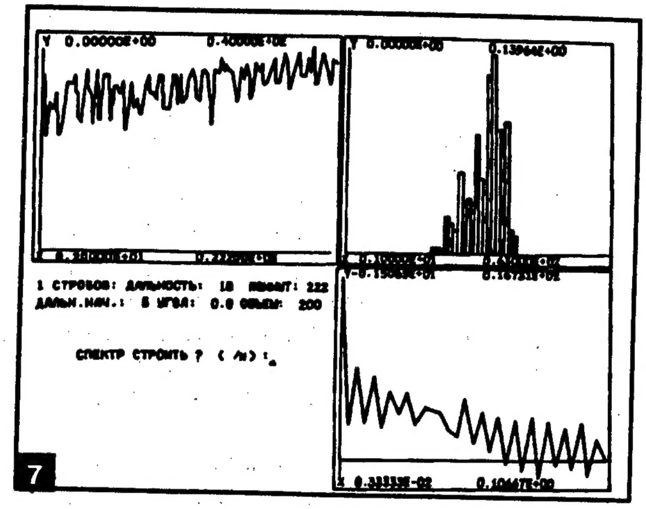


Figure 6 [left]. Results of processing the values of echo signals from the 16th element of distance resolution. Legend: 1 stobes; distance: 18; azimuth: 222. Initial distance: 16; angle: 0.0; volume: 200. Construct correlation function? (/N):

Figure 7 [right]. Results of processing the values of echo signals from the 5th element of distance resolution. Legend: 1 strobes; distance: 18; azimuth 222. Initial distance: 5; angle: 0.0; volume: 200. Construct spectrum? (/N):

correlation function. As can be seen from the data, reflections from this part of the surface hardly fluctuate. The spectrum of fluctuations is uniform. The histogram of the distribution of amplitudes of the echo signals in Figure 6 has a clear bimodal character, and the spectrum of fluctuations shows a clear maximum around 144 Hz. Conversely, the histogram in Figure 7 is unimodal. The spectrum of fluctuations is close to exponential, and has a small maximum near 144 Hz.

These results indicate the complex character of reflections of radar signals from the mountainous formations due to the presence of several bright "brilliant" points in the resolved volume, exceeding the fluxes and turbulence. We note that no such data has been encountered earlier in the literature.

Conclusion. In our opinion the directions of further development of the complex of hardware and software might be the following:

—development of software to study the fine structure (intra-period fluctuations) of echo signals from radar targets and evaluations of multi-dimensional spatial and temporal characteristics of background reflections (noise);
—increase in the number of channels in the system to record information to evaluate the polarization characteristics of echo signals and to measure the

quadrature components of the radar signal of coherent radar stations;
—inclusion in the system of observations combined with radar channels in different wave ranges, for example optical or infrared, and the execution of studies involving the complex processing of the signals of sensors of a different physical nature.

BIBLIOGRAPHY

1. Tverskoy, G. N., Terentyev, G. K., Kharchenko, I. P. "Imitatory Ekho-signalov sudovykh RLS" [Imitators of the Echo Signals of Ship-Borne Radar Stations]. Leningrad: Sudostroyniye 1973 224 pp.

2. Zherlakov, A. V., Zimin, N. S., Kononov, O. V. "Radiolokatsionnyye sistemy preduprezhdeniya stolknoveniya sudov" [Radar Systems to Prevent Ship Collisions] Leningrad: Sudostroyeniye '984 200 pp.

3. "Morskaya radiolokatsiya" [Marine Radar] Edited V. I. Vinokurov. Leningrad: Sudostroyeniye 1986 256 pp.

4. Kuzmin, S. Z. "Osnovy proyektirovaniya sistem tsifrovoy obrabotki radiolokatsionnoy informatsii" [Foundations of the Design of Systems to Digitally Process Radar Information] Moscow: Radio i svyaz 1986, 352 pp.

5. V. S. Kokorin, L. S. Kridiner, A. A. Popov, M. M. Khokhlov. Trends for the Development of Interactive Computing Complexes. MIKROPROTSESSORNYYE SREDSTVA I SISTEMY 1986 No 4 pp 11-15.

6. Glushkova, G. G., Ivanov, Ye. A. The Elektronika microcomputer family. MIKROPROTSESSORNYYE SREDSTVA I SISTEMY 1986 No 4 pp 7-10.

7. "MikroEVM: V 8 kn.: Prakt. posobiye" [The Microcomputer in 8 volumes. A Practical Guide] Edited L. N. Presnukhin. "Kn. 1. Semeystvo EVM 'Elektrnonika 60'" [Book 1 The Elektronika 60 family of computers]. I. L. Talov, A. N. Solovyev, V. D. Borisenko. Moscow: Vyssh. shk. 1988 172 pp.

8. "Interfeysy sistem obrabotki dannykh" [Interfaces of Data Processing Systems] Edited A. A. Myachev. Moscow: Radio i svyaz 1989 416 pp.

9. "Bystrodeystvuyushchiye integralnyye mikroskhemy TsAP i ATsP i izmereniye iz parametrov" [Fast ADC and DAC Integrated Microcircuits and Measurement of Their Parameters]. Edited. A.-I. K. Martsinkyaichus, E.-A. K. Bogdanskis. Moscow: Radio i svyaz 1988 224 pp.

10. Samofalov, K. G., Viktorov, O. V., Kuznyak, A. K. "Mikroprotsessory" [Microprocessors] Kiev: Tekhnika 1986 227 pp.

11. Kornilyev, E. A., Prokopenko, I. G., Chuprin, V. M. "Ustoychivyye algoritmy v avtomatizirovannykh sistemakh obrabotki informatsii" [Stable Algorithms in Automated Systems of Information Processing] Kiev: Tekhnika 1989 224 pp.

12. Tyuki, D. "Analiz rezultatov nablyudeniy" [Analysis of the Results of Observation] Moscow: Mir 1981 267 pp.

13. Khan, G., Shapiro, S. "Satisticheskiye modeli v inzhenernykh zadachakh" [Statistical Models in Engineering Problems] Moscow: Mir 1969 397 pp.

14. "Sudovyye sredstva avtomatizatsii preduprezhdeniya stolknoveniy sudov" [Ship-borne Means of Automating the Prevention of Ship Collisions] Yu. G. Zurabov, R. N. Chernyayev, Ye. V. Yakshevich, V. Ya. Yalovenko. Moscow: Transport 1985 264 pp.

15. Kononov, O. V. Study of the Characteristics of Radar Detection of Above Water Objects. "Sudostroyeniye na morskom flote" [Shipbuilding in the Navy] Moscow: TsRIA Morflot 1982 pp 107-109.

16. Bronshteyn, R. A., Yevtekhov, A. A. Electronic Disk for the Elektronika-60 Computer, DVK [not further expanded] with Energy-Independent Information Storage. MIKROPROTSESSORNYYE SREDSTVA I SISTEMY 1988 No 2 pp 82-84.

17. "Sudovyye radiolokatsionnyye stantsii. Atlas" [Ship-borne Radar Stations. Atlas]. Edited A. M. Bayrashevskiy. Moscow: Transport 1986 142 pp.

- END -

22161
55

NTIS
ATTN: PROCESS 103
5285 PORT ROYAL RD
SPRINGFIELD, VA 22161

Foreign Broadcast Information Service (FBIS) and Joint Publications Research Service (JPRS) publications contain political, military, economic, environmental, and sociological news, commentary, and other information, as well as scientific and technical data and reports. All information has been obtained from foreign radio and television broadcasts, news agency transmissions, newspapers, books, and periodicals. Items generally are processed from the first or best available sources. It should not be inferred that they have been disseminated only in the medium, in the language, or to the area indicated. Items from foreign language sources are translated; those from English-language sources are transcribed. Except for excluding certain diacritics, FBIS renders personal and place-names in accordance with the romanization systems approved for U.S. Government publications by the U.S. Board of Geographic Names.

Headlines, editorial reports, and material enclosed in brackets [] are supplied by FBIS/JPRS. Processing indicators such as [Text] or [Excerpts] in the first line of each item indicate how the information was processed from the original. Unfamiliar names rendered phonetically are enclosed in parentheses. Words or names preceded by a question mark and enclosed in parentheses were not clear from the original source but have been supplied as appropriate to the context. Other unattributed parenthetical notes within the body of an item originate with the source. Times within items are as given by the source. Passages in boldface or italics are as published.

SUBSCRIPTION/PROCUREMENT INFORMATION

The FBIS DAILY REPORT contains current news and information and is published Monday through Friday in eight volumes: China, East Europe, Soviet Union, East Asia, Near East & South Asia, Sub-Saharan Africa, Latin America, and West Europe. Supplements to the DAILY REPORTs may also be available periodically and will be distributed to regular DAILY REPORT subscribers. JPRS publications, which include approximately 50 regional, worldwide, and topical reports, generally contain less time-sensitive information and are published periodically.

Current DAILY REPORTs and JPRS publications are listed in *Government Reports Announcements* issued semimonthly by the National Technical Information Service (NTIS), 5285 Port Royal Road, Springfield, Virginia 22161 and the *Monthly Catalog of U.S. Government Publications* issued by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

The public may subscribe to either hardcover or microfiche versions of the DAILY REPORTs and JPRS publications through NTIS at the above address or by calling (703) 487-4630. Subscription rates will be provided by NTIS upon request. Subscriptions are available outside the United States from NTIS or appointed foreign dealers. New subscribers should expect a 30-day delay in receipt of the first issue.

U.S. Government offices may obtain subscriptions to the DAILY REPORTs or JPRS publications (hardcover or microfiche) at no charge through their sponsoring organizations. For additional information or assistance, call FBIS, (202) 338-6735,or write to P.O. Box 2604, Washington, D.C. 20013. Department of Defense consumers are required to submit requests through appropriate command validation channels to DIA, RTS-2C, Washington, D.C. 20301. (Telephone: (202) 373-3771, Autovon: 243-3771.)

Back issues or single copies of the DAILY REPORTs and JPRS publications are not available. Both the DAILY REPORTs and the JPRS publications are on file for public reference at the Library of Congress and at many Federal Depository Libraries. Reference copies may also be seen at many public and university libraries throughout the United States.